

PVP 225 ptas, incluido IVA y sobretasa aérea Canarias
MSX EXTRA
LA PRIMERA REVISTA DE MSX DE ESPAÑA
N.º 28 Febrero 1987 - PVP 225 ptas. (incluido IVA)
DEL HARD AL SOFT
Elementos básicos de los sistemas de comunicaciones
MITSUBISHI ML-30 FD
Trabajamos con esta unidad de disco y pusimos a prueba su velocidad
PROGRAMAS
Iva
Tokens
Trebol
OVNI
Todos ellos con Test de listados
TRATAMIENTO DE FICHEROS
Describimos en qué consisten y cómo se usan.
GIROMATICA
Profundizamos en el arte de los gráficos rotacionales
CALL VIII
Proponemos el modo de mejorar la programación
BIT BIT
King Leonard
Le Mans 2
Test de Listados
Dragonworld
Merlín
MSX
AMSTRAD
ZX Spectrum

CURSO DE INGLES

The Gruneberg Linkword Language System es un sistema, para enseñanza de idiomas, más rápido y fácil que los métodos convencionales aplicados actualmente.
En poco tiempo, máximo 20 horas, te enseñará un vocabulario de 400 palabras y adquirirás unas buenas nociones de gramática. Esto te permitirá entender y ser entendido en tus viajes a lugares de habla inglesa o en tus contactos con personas que se expresen en ese idioma.
Por otra parte, el Sistema Plus Data, consigue que el ordenador se convierta en un perfecto profesor que te explicará, orientará y corregirá, manteniendo en todo momento un "diálogo" interactivo de resultados sorprendentes.



–L. Taylor. **"POPULAR COMPUTER WORLD":**
"Quedé francamente atónito al comprobar la efectividad de la sugestión de imágenes como elemento de ayuda a la retención..."

–**"PERSONAL COMPUTER WORLD":**
"Un suceso fuera de serie..."

–Bill Barnet. **"COMPUTER CHOICE":**
"De todos los paquetes para aprender idiomas éste es el más interesante..."

A LA VENTA EN GALERIAS PRECIADOS Y EL CORTE INGLES



Programas de EAO para EGB.
Cursos de Basic, Cobol, etc. AUTODIDACTAS.

Nombre ....
Apellidos ....
Dirección ....
Población ....
D.P. .... Tlno. ....
Forma de pago: Reembolso ☐ Giro postal ☐ Envío talón ☐

☐ Curso de Inglés 1.ª parte. 10 lecciones Linkword. (Cinta) P.V.P. 6.900.-Ptas.
☐ Curso de Inglés 1.ª parte. 10 lecciones Linkword. (3,5"-Disk) P.V.P. 7.900.-Ptas.
ENVIAR ESTE CUPON A: PLUS DATA, S.A. C/. GRAN VIA, 661 pral. 08010-Barcelona. Tel. 246 02 02

Editorial

LAS PATAS CORTAS

El rumor como la mentira tiene patas cortas, dice el dicho. El rumor que nos concierne corrió como reguero de pólvora el pasado año y consistía en el fulminante éxito de aparatos como el Amstrad y el Spectrum Plus. Tal éxito implicaba la derrota por K.O. del MSX. Nada de esto era cierto, pero muchos lectores se habrán sentido incómodos al ver cómo otro sistema desplazaba a aquél en el que tanto confiaba él y nosotros. Sin embargo, a la larga todo se esclarece y es el mismo público el que determina el alcance y persistencia de un producto en el mercado.
Al cierre del ejercicio anterior las ventas de aparatos MSX –según información de los principales fabricantes y distribuidores del sistema–, han sido espectaculares, llegando a agotar las existencias en muchos casos. La situación actual de la norma es más saludable que nunca y permite suponer que las presuntas competencias de otros aparatos, cuyas marcas oscilan entre un nombre y otro, según los vaivenes económicos de sus fabricantes, no es tan real como parece.
El panorama como vemos es optimista para la norma MSX, sobre todo porque sus niveles de compatibilidad crecen (IBM acaba de lanzar un nuevo aparato con diskettes de 3,5 pulgadas) facilitando su asentamiento. De todos modos este optimismo se ha de seguir alimentando con hechos. Por un lado los fabricantes creando más software de calidad y periféricos y facilitando una mayor información de sus productos, tanto a través de la publicidad como del periodismo; y por otro, los usuarios manteniendo su fe, haciendo oídos sordos a los rumores. Entre unos y otros, la prensa especializada sirviendo de vehículo tanto a fabricantes y vendedores, como a usuarios. Es cuestión de organizar bien el equipo.

MANHATTAN TRANSFER, S.A.



SUMARIO

AÑO III N.º 28 FEBRERO 1987
P.V.P. 225 ptas. (Incluido IVA y sobretasa aérea Canarias)
Aparece los días 15 de cada mes.

MSX EXTRA ES EDITADA POR MANHATTAN TRANSFER, S.A.
Director Editorial: Antonio Tello Salvatierra.
Director Ejecutivo: Birgitta Sandberg.
Redactor Jefe: Javier Guerrero. **Redactores:** Claudia T. Helbling, Silvestre Fernández y Rubén Jiménez. **Colaboradores:** Angel Toribio, Joaquín López, J.M. Climent Parcet. **Departamento de programación:** Juan C. González. **Diseño:** Félix Llanos. **Grafismo:** Juan Núñez, Carles Rubio. **Suscripciones:** Silvia Soler. **Redacción, Administración y Publicidad:** Roca i Batlle, 10-12. 08023 Barcelona. Tel. (93) 211 22 56.
Fotomecánica y Fotocomposición: Ungraf, S.A. Pujadas, 77-79. 08005 Barcelona.
Imprime: Grefol, Polig. II Lafuensanta Parc. 1 Móstoles (Madrid)
Distribuye: GME, S.A. Plaza de Castilla 3, 15.º E. 2. 28046 Madrid

Input



TONOS DE COLOR

Me gustaría saber cómo se hace, por ejemplo, un cubo pintado por todas sus caras de forma que se note la diferencia de color entre cara y cara. Si se puede hacer, claro.

Oscar Vázquez Viladoms
Sant Quintí de Mediona
(BARCELONA)

Supongo que a lo que te refieres es a cómo conseguir diferentes tonalidades de un mismo color. Esto se puede conseguir, en los MSX de segunda generación, con el comando PALETTE, que permite la definición de la paleta de colores a nuestro gusto.

En los MSX de primera generación resulta un poco más complicado, aunque también es posible. La solución es simple. Hay que generar una trama de puntos de diferente color, con una densidad proporcional al tono de color que desees.

Imagina que quieres hacer una línea de color naranja. Para ello no tienes más que dibujar un punto amarillo, y uno rojo, unc amarillo, y otro rojo, y así hasta acabar de dibujar la línea. Para conseguir un naranja más claro debes colocar dos puntos amarillos entre cada punto rojo, etc.

Este interesante truco permite generar colores no existentes habitualmente en los MSX, como son el marrón o el naranja. Hemos de avisar que en la mayoría de los casos los colores se complementan de forma que no se aprecia que estén formados por conjunción de otros; pero hay que probar todas las combinaciones para llegar a dominar este truco, muy utilizado por los programas de juegos comerciales.

FOTOS DE PANTALLAS DE TV O MONITOR:

¿Pueden decirme la forma de hacer buenas fotos de pantallas de TV o monitor?

Manuel Rodríguez López
CACERES

El resultado de las fotos realizadas sobre una pantalla depende mucho de las condiciones en que ésta sea tomada. Te enumeramos a continuación algunas normas a tener en cuenta a la hora de realizar las fotos.

En primer lugar intenta evitar todo tipo de reflejos en la pantalla. Para ello deja a oscuras la habitación en la que se encuentre la pantalla. Debido a esto debes tomar también la foto sin flash.

Ajusta los controles de brillo y contraste del monitor de modo que la imagen adquiera un colorido brillante; pero no exagerado.

Ajusta la cámara a una distancia de unos 40 cm. de la pantalla. Las fotos resultarán mucho más claras si las fijas con un trípode.

Selecciona una velocidad entre 1/30 y 1/60 de segundo y coloca el diafragma entre 2 y 4. Estas dos últimas especificaciones están indicadas para una película de sensibilidad normal, de 100 ASA.

Ya sólo te queda disparar. Como te hemos dicho antes, los resultados pueden variar dependiendo de las condiciones en que realices la fotografía. Asimismo, variarán en caso de que utilices un macro o algún otro objetivo especial.

VIDEO INVERSO EN LOS MSX-2

Poseo un ordenador PHILIPS MSX-2 desde hace unos meses y me estoy dedicando al estudio serio de la informática con vistas a hacerme mis propios programas. Todo ello en BASIC. Al entrar a mi aparato me encontré con una pequeña pega: ¿cómo poder conseguir textos en VIDEO INVERSO en SCREEN 0 y WIDTH 80?

Manuel Renedo Gil
Sta. Cruz de Tenerife

En primer lugar hemos de felicitarte por tu empeño en desarrollar tus propios programas de utilidad, y animamos a todos a que se dediquen a la programación todo cuanto puedan ya que és ésta la única forma de utilizar al máximo las posibilidades de vuestro ordenador.

Ya comentamos en otra ocasión cómo conseguir vídeo inverso en los MSX (de primera generación), por lo que no trataremos en detalle cómo funciona el SET de CARACTERES en la VRAM. Simplemente incluimos a continuación el programa que imprime caracteres en vídeo inverso. Para ello debes imprimir cada carácter más el código 128 (observa el listado). Recuerda que pierdes los caracteres gráficos. Puedes mejorar, sin embargo, el programa con poco que te lo propongas.

```
10 FOR X= 4096 TO 5119
20 VPOKE X + 1024, VPEEK
(X) XOR 255
30 NEXT X
40 A$ = "Este mensaje aparece en vídeo inverso".
50 FOR X = 1 TO LEN (A$)
60 PRINT CHR$ (128+ASC
(MID$(A$, X1)));
70 NEXT X
```

¿REGLAS DE NUESTROS PROGRAMAS?

Quisiera que me respondiérais a algunas preguntas sobre vuestro concurso de programas.

—¿Se pueden mandar programas para unidades de disco?

—¿Existe algún límite en extensión?

Carlos Márquez Reyes
ALMERIA

Hemos de deciros, a ti, y todos los interesados en nuestro concurso de programas, que no hay ninguna limitación temática en nuestro concurso. Así, tampoco hay limitación de memoria, aceptándose por tanto, los programas de ordenadores de la segunda generación.

LA BARRERA DE LAS 40 COLUMNAS

A causa de que estoy realizando un programa, necesitaría reducir el tamaño de los caracteres. He leído que para ello es necesario ocupar áreas de memorias reservadas o especificadas al MSX-2, consiguiendo así 64 caracteres por línea. ¿Es cierto esto? ¿Cómo puedo conseguirlo?

José Florencio Sánchez Rúa
Vigo (PONTEVEDRA)

En primer lugar no comprendemos por qué nos hablas de MSX-2 (segunda generación de MSX), ya que estos cuentan con pantalla de 80 columnas. Para conseguirlo no hay más que teclear WIDTH 80.

Suponemos que te refieres a las zonas de memoria del MSX-DOS (sistema operativo de disco MSX). La respuesta a tu pregunta es sencilla; pero no tan sencilla de llevar a cabo.

En primer lugar, debes saber que el VDP (chip de pantalla) de los MSX de primera generación sólo permite trabajar con 40 columnas. Por lo tanto es imposible hacerlo trabajar con más de 40 columnas.

Si deseas utilizar 64 columnas sólo te caben dos opciones: cambiar el VDP, o bien no utilizar caracteres.

No utilizar caracteres quiere decir que no se pueden utilizar los caracteres normales (los que utiliza el comando PRINT), sino que debemos definir un set complementario y dibujar los caracteres.

En los MSX de primera generación es posible simular 64 columnas en SCREEN 2 si dibujamos cada uno de los caracteres como una trama de 4 × 8 caracteres. Este efecto se puede lograr fácilmente mediante el macrocomando DRAW; pero es muy lento, por lo que se precisa de una rutina en ensamblador.

Nuestro equipo de programadores ya está trabajando, desde hace algún tiempo en la generación de esta rutina; por lo que esperamos que, no dentro de demasiado tiempo, podamos incluir dicha rutina en nuestras páginas.

Evidentemente, la mayoría de nuestros lectores no tienen unidad de disco, por lo que la mayoría de los programas que publiquemos serán para ordenadores con cassette; pero resulta interesante que nos enviéis programas de todo tipo, para así poder satisfacer a todos los lectores de una forma equitativa.

Evidentemente, los criterios seguidos por nuestro jurado son múltiples, y el hecho de que no hayamos publicado anteriormente programas en disco, ha sido que no hemos recibido programas de la suficiente calidad. Sí, sin embargo, hemos publidado varios artículos dedicados a este tema, al MSX-2, y los otros muchos apartados que podemos encontrar dentro del mundo de los MSX.



SISTEMA OPERATIVO MS-DOS

Tengo un Spectravídeo SV-728. ¿Es posible trabajar con el sistema operativo MS-DOS?

Mikel Amunarrit
Fuenterravía
(GUIPUZCOA)

Los ordenadores MSX no soportan el sistema operativo MS-DOS, ya que este sistema operativo se encuentra sólo disponible en ordenadores de 16 bits, como el IBM PC y compatibles. Los MSX cuentan con una adaptación de este sistema operativo, que es lo que conocemos como MSX-DOS, y que ha sido desarrollado por Microsoft, que también desarrolló el MS-DOS.

Lamentablemente, los programas que corren bajo sistemas operativos MS-DOS no lo hacen sobre MSX-DOS; pero no por culpa del sistema operativo, ya que el MSX-DOS mantiene la compatibilidad de las llamadas de la BIOS, sino por culpa del Z-80A, incapaz de ejecutar las instrucciones del Intel 8088. o del Intel 8086.

AMPLIACION DE MEMORIA EN EL HB-101/P

Poseo un Sony HB-101P con 32 Kb de RAM (+ 16 Kb de VRAM), ROM de 48 Kb (incluidos 16 Kb del Personal Data Bank). Ante la imposibilidad de cargar programas de más de 32 Kb, adquirí un cartucho de ampliación de memoria. La ampliación permite cargar programas de 64 Kb como Alien 8; pero sin embargo se "cuelga" con otros como NightShade. ¿Por qué sucede esto? ¿Cómo se puede solucionar?

Rafa Flores Yoldi
Pamplona (NAVARRA)

El problema que comentas se debe a la especial configuración de la memoria que tiene tu ordenador, y a que los programas que comentas (los que no funcionan), no hacen un uso correcto de las rutinas de paginación de memoria. En primer lugar hemos de agradecerte que nos envíases los mapas de memoria de tu ordenador (que reproducimos a continuación), ya que nos ha facilitado en gran medida el poder responder a tu carta. Aprovechamos estas líneas para pedir a otros consultantes que nos envíen el máximo de información junto con sus dudas. No nos basta con saber que un programa no funciona para saber cuál es la posible causa.

El problema es fácil de localizar y modificar si tienes unos ligeros conocimientos de lenguaje ensamblador. Muchos programas presuponen que los 64 Kb del ordenador están en un mismo SLOT. Esto ocurre en la mayoría de los MSX; pero no en todos. Se utiliza una rutina similar a la que incluimos a continuación.

```
(1) IN A, (#A8)
(2) AND FO
(3) LD B, A
(4) SHR A
    SHR A
    SHR A
    SHR A
    SHR A
(5) ADD A, B
(6) OUT (#A8), A
(7) RET
```

En primer lugar carga el banco que actualmente se esté utilizando. El programa se basa en que actualmente el programa está en RAM. En la línea 2 (evidentemente los números están solo para aclarar la explicación) eliminamos los 4 bit menos significativos del número de banco. Es decir, nos quedamos sólo con el código de banco de las dos zonas de RAM (de los 8 bits, la mitad superior, que corresponden a los 32Kb superiores de la memoria, ocupada por la RAM). En la línea 4 desplazamos 4 lugares hacia la derecha el código de los bancos de RAM, y en la línea 5 le sumamos los 4 bits superiores. Hemos conseguido de este modo una copia de los 4 bits superiores en los 4 bits inferiores. Si al comenzar el programa en la parte superior había RAM, la habrá ahora en la zona inferior. En la línea 6 conectamos el banco calculado y en la 7 acabamos la rutina.

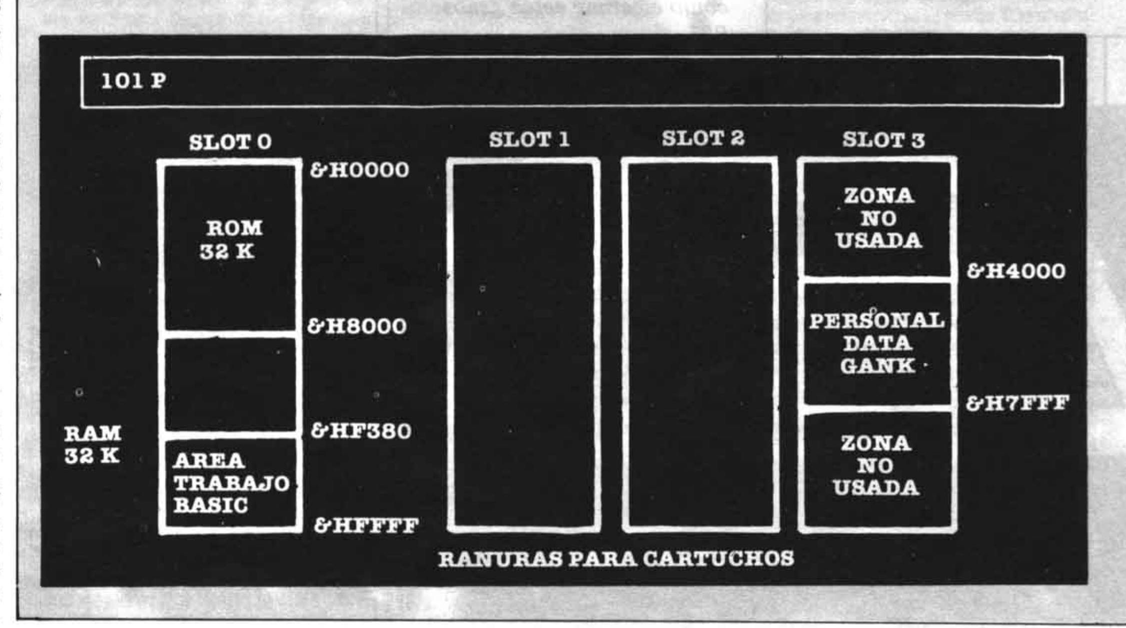


Como ves, esta rutina no funcionará en tu aparato. Solucionar este problema no es difícil. En primer lugar debes localizar en el programa (con ayuda de un desensamblador) una rutina similar a esta. Y sustituirla por un:

LD A # 50
OUT (#A8), A si utilizas el SLOT 1 para conectar la ampliación o bien

LD A, #A0
OUT (#A8), A si utilizas el SLOT 2 para conectar la ampliación.

Esperamos que con esto consigas hacer funcionar todos tus programas.



Input — Output

MEMORIA EN OTROS LENGUAJES

En los MSX sólo disponemos de 28 Kb si trabajamos en BASIC, y dispondremos del resto de la memoria si trabajamos en código máquina.

Si trabajamos en COBOL, al compilar el programa, es decir, traducirlo a código máquina; ¿podremos disponer de toda la memoria?, es decir, ¿en un MSX-2 de 128 Kb podemos hacer un programa de 100Kb?

¿Es verdad que los discos grabados en otros MSX no cargan en el Spectravídeo X'PRESS?

Javier Uroz y José A. Lomas
Terrassa (BARCELONA)

Como bien decís, en BASIC sólo contamos con 28 Kb de memoria (24 si está conectada la unidad de disco). Para utilizar el resto de la memoria debemos trabajar en ensamblador. Como ya hemos comentado en anteriores ocasiones, el Z-80, chip que controla los MSX de ambas generaciones, sólo puede trabajar con 64 Kb de memoria. Esta es la memoria máxima accesible desde el sistema operativo. Debemos decirte, muy a pesar nuestro, que probablemente no puedas superar los 64 Kb aunque programes en COBOL. Unicamente te podrías superar esta barrera si tu compilador de COBOL fuese capaz de manejar los bancos de memoria. No conocemos ningún compilador de COBOL MSX que maneje los bancos; pero al ser éste un programa de poca difusión pueden existir otras versiones no conocidas por nosotros.

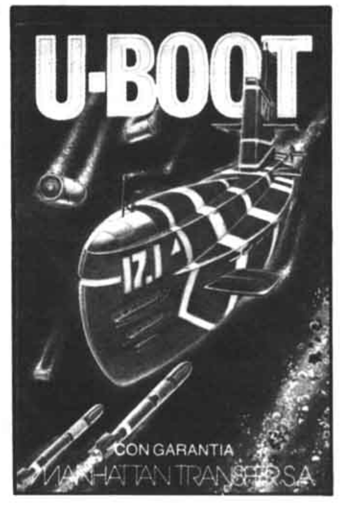
Tu segunda pregunta es un poco más complicada de responder. Los discos «normales» son compatibles, tanto en el Spectravídeo X'Press, como en cualquier otro MSX. De esta manera, todo programa puede pasarse de un MSX a otro. Sin embargo, algunos MSX, como los nuevos modelos de X'Press o los MSX-2, incorporan unidades de alta capacidad. Por ejemplo, la unidad de alta capacidad de los MSX-2 permite almacenar 720Kb en un solo disco. Evidentemente, este disco no puede ser leído con un MSX de la primera generación, que sólo lee 360 Kb. Sin embargo, con el MSX-2 puedes grabar también discos de 360 Kb (las unidades de alta capacidad pueden grabar en los dos formatos). En resumen: si deseas que un disco pueda ser leído por otro MSX debes grabarlo en formato de 360 Kb. Dirígete al manual del DISK-BASIC o del MSX-DOS para saber como efectuar estas grabaciones.



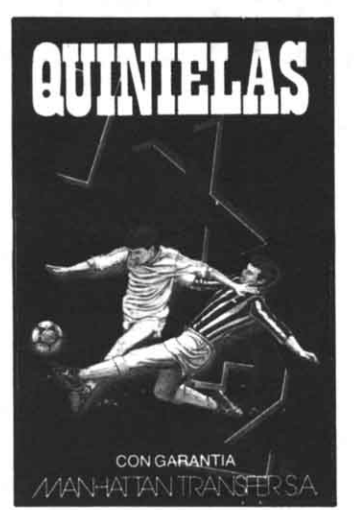
NIDOS A msxclub de CASSETTES



KRYPTON. La batalla más audaz de las galaxias en cuatro pantallas y cuatro niveles de dificultad. Un juego cuya popularidad es cada vez más grande entre los usuarios del MSX. PVP. 500 Ptas.



U-BOOT. Sensacional juego de simulación submarina en la que tienes que demostrar tu pericia como capitán de un poderoso submarino de guerra. Panel de mandos, sonar, torpedos, etc. PVP. 700 Ptas.



QUINIELAS. El más completo programa de quinielas con estadística de la liga, de los aciertos, etc. e impresión de boletos. Acertar no siempre es cuestión de suerte. PVP. 700 Ptas.



SNAKE. Entretenido y muy divertido juego en el que Snake procura comer unos números que la engordan. Tanto las murallas que la rodean como su larga cola pueden ser mortales para ella. PVP. 600 Ptas.



EL SECRETO DE LA PIRAMIDE. Atrevido juego de aventuras a través de los misterios y peligros que encierran los laberínticos pasillos de una pirámide egipcia. ¡Atrévete si puedes! PVP. 700 Ptas.



STAR RUNNER. Conviértete en el audaz piloto interestelar y lucha a muerte, a través del hiperespacio, contra las defensas del tirano Daurus. Dos pantallas y cinco niveles de dificultad. PVP. 1.000 pts.



FLOPPY, El Preguntón. Un verdadero desafío a tus conocimientos de Geografía e Historia española. Floppy no perdona y te costará mucho superarlo. PVP. 1.000 Ptas.



MAD FOX. Un héroe solitario es lanzado a una carrera a vida o muerte por un desierto plagado de peligros. Conseguir el combustible para sobrevivir es su misión. Diez niveles de dificultad. PVP 1.000 pts.

Si quieres recibir por correo certificado estas cassettes garantizadas recorta o copia este boletín y envíalo hoy mismo:

Nombre y apellidos: ....

Dirección: ....

Población: .... **CP** .... **Prov.** .... **Tel.:** ....

☐ KRYPTON	Ptas. 500,–	☐ SNAKE	Ptas. 600,–	☐ FLOPPY	Ptas. 1.000,–
☐ U BOOT	Ptas. 700,–	☐ EL SECRETO DE LA PIRAMIDE	Ptas. 700,–	☐ MAD FOX	Ptas. 1.000,–
☐ QUINIELAS	Ptas. 700,–	☐ STAR RUNNER	Ptas. 1.000,–	☐ VAMPIRO	Ptas. 800,–
☐ HARD COPY	Ptas. 2500,–	☐ TEST DE LISTADOS	Ptas. 500,–	☐ SKY HUNTER	Ptas. 1.000,–
☐ LORD WATSON	Ptas. 1.000,–				

Gastos de envio certificado por cada cassette .... Ptas. 70,– Remito talón bancario de Ptas. .... a la orden de Manhattan Transfer, S.A.

ATENCION: Los suscriptores tienen un descuento del 10% sobre el precio de cada cassette.

IMPORTANTE: Indicar en el sobre MSX CLUB DE CASSETTES. ROCA I BATLLE, 10-12 BAJOS. 08023 BARCELONA

Para evitar demoras en la entrega es imprescindible indicar nuestro nuevo código postal.

NUESTRAS CASSETTES NO SE VENDEN EN QUIOSCOS. LA UNICA FORMA DE ADQUIRIRLAS ES SOLICITANDOLAS A NUESTRA REDACCION. ¡NO SE ADMITE CONTRA REEMBOLSO!

MITSUBISHI ML-30 FD UNIDAD DE DISCOS

UN RETO A LA VELOCIDAD

Incluimos en nuestra sección BANCO DE PRUEBAS el test realizado a la unidad de discos Mitsubishi ML-30FD. Respondemos de este modo (y con otros test que estamos realizando) a las demandas de algunos lectores sobre uno de los periféricos más importantes de los MSX.

PRIMERA IMPRESION

Nada más desembalar la ML-30FD nos damos cuenta de que estamos ante una unidad singular. Su exterior, atractivo y funcional, nos da una veraz impresión de las posibilidades de este periférico.

En primer lugar destaca su posición vertical, frente a la posición horizontal de la gran mayoría de unidad de disco. Este modo de disponer la unidad, aparte de original entre los MSX, resulta extremadamente práctico, ya que nos permite un mayor aprovechamiento del espacio de trabajo (por lo general reducido).

EL FRONTAL DE LA UNIDAD

La unidad presenta una ergonomía bastante buena (recordamos que la ergonomía es la mayor o menor capacidad de un aparato para ser usado por los humanos: facilidad, comodidad, practicidad, etc). La única duda que aborda al que ya ha trabajado con otras unidades es cómo se introduce el disco. En una unidad convencional esto es claro: la parte superior del disco hacia arriba y la inferior hacia abajo. Pero en esta unidad, no se sabe si colocar la parte superior a la izquierda o a la derecha. Un rápido vistazo al manual nos saca de dudas: la parte superior hacia la izquierda.

En el frontal se distinguen varias partes importantes: el orificio para introducir el disco 1, el botón de expulsión del disco, y el led de funcionamiento del disco. También encontramos en el frontal el interruptor de encendido y el led que indica que la unidad se halla conectada.



Pero el punto más importante a destacar del frontal de esta unidad, es que incorpora un espacio dedicado a poder incorporar una segunda unidad. Esta segunda unidad de disco se suministra por separado; pero como la alimentación, la carcasa y el controlador de disco ya están incorporados en la primera unidad, su precio es claramente inferior. Esta característica la hace especialmente atractiva a aquellos usuarios que deseen utilizar su MSX con dos unidades de disco de forma simultánea.



Tabla de velocidades de la unidad de disco Mitsubishi ML-30FD

Grabación de 1024 fichas de un carácter	**33 seg.**
Grabación de un bloque de 16 Kb	**6 seg.**
Lectura de un bloque de 16 Kb	**5 seg.**
Abrir y cerrar un fichero 100 veces	**1 min. 21 seg.**

Especificaciones

Generalidades	
Consumo de energía	**16 W**
Condiciones ambientales	**5 - 35°C, 20 - 80% HR**
Dimensiones (AxGxH)	**86 × 270 × 166 mm**
Peso	**2,5 kg**
Unidad de disco microflexible	
Capacidad de almacenamiento	
No ajustado a formato	**1M byte**
Ajustado a formato	**720K bytes**
Velocidad de transmisión	**250K bits/seg.**
Tiempo de acceso	
Pista-a-pista	**12 ms**
Acceso promedio	**331 ms (incluyendo anillo de búsqueda y fijo)**
Tiempo de anillo fijo	**15 ms**

EL PANEL TRASERO

El panel trasero demuestra una sencillez extremada. En él sólo encontramos el cable de conexión a la red eléctrica y el cable que la comunica con el controlador de disco, instalado en una ranura de cartucho del ordenador.

Resulta, por tanto, muy sencillo instalar la unidad, aunque no se tenga ningún conocimiento previo. Para instalar la unidad sólo hay que conectarla al controlador con un sólo cable y enchufar la unidad al enchufe más próximo.

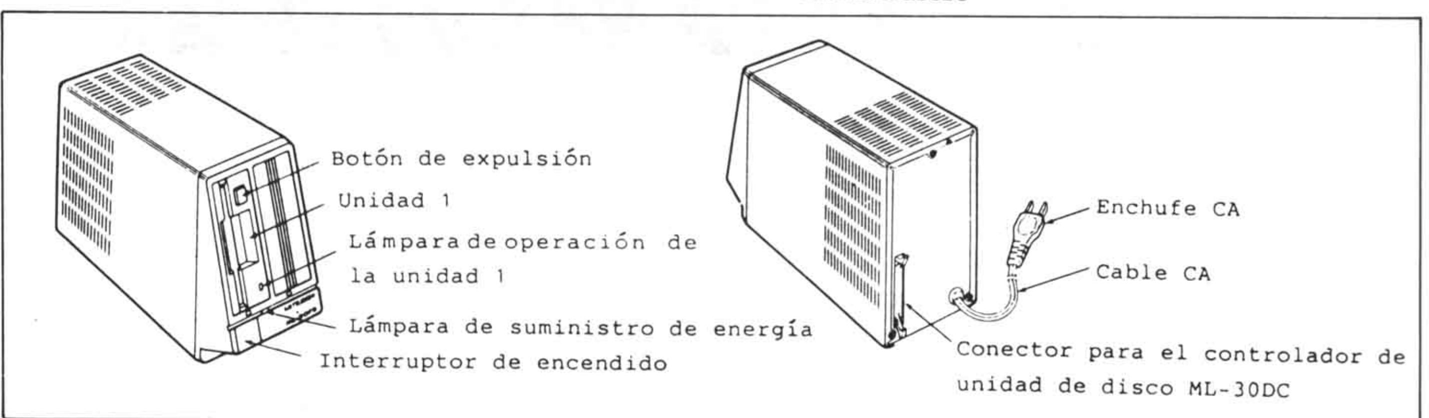
EL CONTROLADOR DE DISCOS

El controlador de discos es el aparato encargado precisamente, como su nombre indica, de controlar a la unidad de disco. Este controlador, como la mayoría de los existentes sobre MSX es capaz de controlar dos unidades de disco. En el caso de este modelo, la segunda unidad debe instalarse en la carcasa de la primera, en el espacio que tiene reservado para ello.

El controlador de discos de la unidad de Mitsubishi es el ML-30DC. Este controlador, como la mayoría de los preparados para MSX se conecta al ordenador por medio de los conectores de cartucho. Uno de los inconvenientes que hemos encontrado al controlador (compartido por otros muchos controladores MSX) es su elevada altura, de modo que, si tenemos colocado el televisor o monitor detrás del ordenador inevitablemente tapa parte de la imagen.

El tipo de cable utilizado para la conexión entre el controlador y la unidad de disco es de tipo banda. Este tipo de cable tiene la ventaja de ser muy flexible, y difícil de romper, pero, a cambio, resulta

Panel delantero **Panel trasero**



más incómodo. Mitsubishi ha optado en este caso por la seguridad en su unidad de disco.

ESPECIFICACIONES TECNICAS

La unidad cuenta con una capacidad de almacenamiento en disco de 1 Mb, que queda reducido a 720Kb al formatear el disco. La velocidad de transmisión es de 250 Kbits por segundo, mientras que el tiempo de acceso pista a pista es de 12 milisegundos.

La densidad de almacenamiento, como en el resto de las unidades MSX es de 8.700 bits por pulgada. Como detalle curioso sabed que la unidad gira a una velocidad de 300 revoluciones por minuto.

Para una mayor información os adjuntamos una tabla de características técnicas.

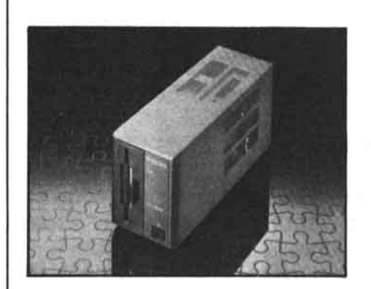
VELOCIDAD DE LA UNIDAD

Existen muchos parámetros utilizados para medir la velocidad de una unidad de disco; pero, indudablemente, el medio más eficaz para medir la velocidad de una unidad es realizar un BENCHMARK, (banco de pruebas) consistente en cronometrar diversas operaciones con los resultados a nuestros bancos de pruebas.

3.º GRAN PROGRAMA



CONCURSO DEL AÑO



CREA Y ENVIANOS TU PROGRAMA. HAY PREMIOS PARA TI Y PARA LOS QUE TE VOTEN. CADA MES PUBLICAREMOS MAS DE UN GANADOR QUE OPTARA UNA FABULOSA UNIDAD DE DISCO

BASES

1 - Podrán participar todos nuestros lectores cualquiera sea su edad, con uno o más programas escritos en BASIC MSX o código Máquina.

2 - Los programas se clasificarán en tres categorías:
A— Educativos
B— Gestión
C— Entretenimientos

3 - Los programas, sin excepción, deberán ser remitidos grabados en cassette virgen, debidamente protegida dentro de su estuche plástico en el que se insertará el cupón-etiqueta que aparece en esta misma página, debidamente rellenado.

4 - No entrarán en concurso aquellos programas plagiados o ya publicados en otras publicaciones nacionales o extranjeras.

5 - Junto a los programas se incluirán en hoja aparte las instrucciones correspondientes, detalle de las variables, ampliaciones o mejoras posibles y todos aquellos comentarios que el autor considere de interés.

6 - Todos los programas han de estar estructurados de modo claro, separando con REM los distintos apartados del mismo.

PREMIOS

7 - MSX EXTRA otorgará los siguientes premios:
AL PROGRAMA MSX EXTRA DEL AÑO
Una Unidad de disco
valorada en más de 80.000 ptas.

8 - Los programas seleccionados por nuestro Departamento de Programación y publicados en cada número de nuestra revista recibirán los siguientes premios en metálico:
Programa Educativo 10.000 pts.
Programa de Gestión 10.000 pts.
Programa de Entretenimiento 6.000 pts.

9 - MSX EXTRA se reserva el derecho de publicar fuera de concurso aquellos programas de reducidas dimensiones que sean de interés, premiando a sus autores.

FALLO Y JURADO

10 - Nuestro Departamento de Programación analizará todos los programas recibidos y hará la primera selección, de la que saldrán los programas que publiquemos en cada número de MSX EXTRA.

11 - Los programas recibidos no se devolverán, salvo que el autor lo requiera expresamente.

12 - La elección del PROGRAMA MSX EXTRA DEL AÑO se hará por votación de nuestros lectores a través de un boletín que se publicará en el mes de octubre de 1987.

13 - El plazo de entrega de los programas finaliza el 15 de noviembre de 1987.

14 - El fallo se dará a conocer en el número del mes de enero de 1988, entregándose los premios el mismo mes.

CORTAR O FOTOCOPIAR

TITULO .... **N.º**

TITULO ....

CATEGORIA
PARA .... **K**
INSTRUCCION DE CARGA

AUTOR:
EDAD:
CALLE: .... **N.º** ....
CIUDAD .... **DP** ..... **TEL.:** ....
N.º DE RECEPCION

REMITIR A:
CONCURSO MSX EXTRA
Roca i Batlle, 10-12 bajos
08023 Barcelona

MEJORAR LA PROGRAMACION

La programación en BASIC tiene sus limitaciones. En estas páginas se muestran algunas posibles soluciones.

El problema del BASIC es que resulta fácil construir programas que funcionan.

Mientras se desarrolla uno de estos, el programador va poniendo "parches" ahí donde son necesarios. El resultado es un programa que, en la mayoría de los casos, funciona mal, aunque a primera vista no lo parezca. Por si ello fuera poco, será muy costoso depurarlo o modificarlo y resultará casi ininteligible para el resto de los mortales.

Sin embargo, esto no implica que todos los programas escritos en BASIC tengan defectos notorios. Es corriente encontrar listados poco optimizados, que consumen mucho tiempo al ser ejecutados y derrochan más memoria de la que deberían. Aquí tienes un ejemplo que ocurre todos los días.

LISTADO 1

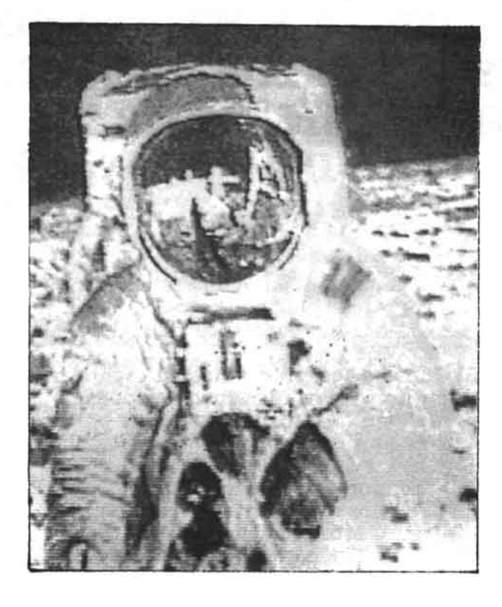
```
10 A=1:B=1:C=1:D=1:E=1:F=1
20 TIME=0
30 FORX=0TO500:'BUCLE DE ESPERA
40 NEXT X
50 PRINTTIME/50
```

Supón que las líneas anteriores forman parte de un programa más largo y que es importante que el bucle contenido en ellas se ejecute lo más rápido posible. Si haces RUN, observarás que el tiempo empleado en el proceso es de 1.66 segundos (o algo similar). Pues bien, prueba a eliminar los dos puntos de la línea 30 y el comentario que le sigue. Al correr el programa, el tiempo de ejecución pasa a ser de 1.08 s.

Se ha conseguido un ahorro sustancial, pero aún puede mejorarse. Elimina la línea 10 y comprobarás que el bucle se completa en sólo 0.98 s. Si además quitas la línea 40 y pones el "NEXT X" al final de la línea 30, el programa no variará pero el tiempo invertido en el bucle pasará a ser de 0.96 segundos.

A continuación, borra la "X" que sigue a "NEXT" y verás como el tiempo se reduce a 0.78 s.

Aún resta una modificación, que es la que consigue un mejor ahorro de tiempo: intercala entre la "X" y el "=" el símbolo "%", correspondiente a las variables enteras. El programa corre ahora en sólo 0.28 segundos, es decir, la sexta parte de lo que invertía el original.



Aquí tienes un resumen de los acontecimientos:

Programa original	1.66
Sin comentario	1.08
Sin línea 10	0.98
Poniendo el NEXT X en la línea 30	0.96
Quitando la X después del NEXT	0.78
Usando una variable entera	0.28

Por todo esto, es recomendable respetar ciertas normas cuando se desea que la ejecución sea lo más rápida posible:

- Las variables que se definen al principio de un programa son las de acceso más rápido.
- Los comentarios influyen negativamente en la velocidad de ejecución.
- Es conveniente situar las rutinas de velocidad crítica al principio del programa.

Usar "NEXT" en favor de "NEXT variable".

- Emplear, en lo posible, variables enteras o, en el peor de los casos, reales, en preferencia a las de doble precisión.

LOS PROGRAMAS "ESPAGUETI"

El abuso de instrucciones condicionales anidadas, puede conducir a un programa enrevesado y confuso. Lo que en el argot se denomina programa "espagueti". Observa la siguiente línea:

10 IF A<2 AND B>0 AND C<1 AND D<1 THEN R=10 ELSE R=0

Este ejemplo no es, ni con mucho, un derroche de confusión. Todos hemos construido programas en los que había líneas bastante peores que la anterior. De cualquier forma, sirve como ejemplo.

Observa como se puede hacer lo mismo empleando algo más corto y, por consiguiente, más claro:

10 R=-10*(A<2)*(B>0)*(C <1)*(D<1)

Sin lugar a dudas, existe cierto desconocimiento sobre los operadores lógicos del BASIC, a pesar de que todos los manuales que acompañan a las máquinas son lo suficientemente claros al respecto. Prueba de ello es que se siguen recibiendo listados con líneas como:

10 IF A <>0 THEN...

Naturalmente se puede conseguir lo mismo con:

10 IF A THEN...

Cosas como éstas no pueden ser tachadas de grandes errores, pero dicen bastante sobre la falta de cuidado (o de preparación) del que los comete.

Sin ir muy lejos, hace unas semanas se recibió un programa que conmutaba el tipo de SPRITE en plena ejecución. Como probablemente sabrás, no es posible emplear la instrucción SCREEN para cambiar el tipo de SPRITE sin borrar todas las figuras definidas. Pues bien, en el listado en cuestión se aprovechaba el hecho de que el registro 1 del VDP determina el tipo de SPRITE, así pues es suficiente con "encender" el bit uno del referido registro para ampliar los SPRITES, si aún no lo estaban. La argucia empleada era esta:

```
10 A$=BIN$(VDP(1))
20 MID$(A$(7,1))="1"
30 VDP(1)=VAL("&B"+A$)
```

La solución es ingeniosa, pero es mucho más simple hacer:

```
10 VDP(1)=VDP(1)OR3
```

Por otra parte, si lo que se desea es cambiar a cualquier tipo y/o ampliación, habrá que usar:

```
10 VDP(1)=No. de TIPO+(VDP(1)AND252)
```

Si no tienes muy claro el funcionamiento de los operadores lógicos, te recomiendo encarecidamente que consultes tu manual, que seguramente contendrá más información de la que se podría dar a través de estas páginas.

INSTRUCCIONES "ON"

Dos instrucciones que no se usan lo que deberían son ON GOTO y ON GOSUB.

Así, es corriente encontrar listados con líneas parecidas a estas:

LISTADO 2

```
1 IF A=1 THEN 10
2 IF A=10 THEN 20
3 IF A=100 THEN 30
4 IF A=1000 THEN 40
5 IF A=10000 THEN 50
```

Observa que todas las líneas anteriores pueden ser sustituidas por una sola:

1 ON 1+LOG(A)/LOG(10) GOTO, 10, 20, 30, 40, 50

Otra instrucción poco utilizada es ON KEY GOSUB.

Supón que estás construyendo un programa en el que el usuario tiene que entrar datos desde el teclado (una agenda, por ejemplo). En este caso será útil que el operador pueda, siempre que lo desee, dejar la operación que estaba ejecutando y solicitar, digamos, una pantalla de ayuda. Está clara la ventaja que supone usar ON KEY GOSUB para detectar las interrupciones y efectuar la función reclamada por el usuario.

Tras la pulsación de una de las teclas de función, es muy probable que la pantalla sea borrada y se aborte la tarea que se estaba realizando en el momento de la interrupción. El problema es que ON KEY GOSUB actúa como subrutina y espera encontrar un RETURN al final. Puesto que suele ser imposible volver al mismo punto desde el que se partió, lo normal es pasar la ejecución a la parte principal (típicamente al menú), usando un RETURN seguido de un número de línea.

Hasta aquí todo va bien, pero hay que tener presente que las llamadas a subrutinas almacenan la línea de retorno en la pila del BASIC. Teclea estas líneas y lo comprobarás:

10 GOSUB 20
20 GOSUB 10

Como ves, si no se desapilan correctamente todas las llamadas a subrutinas se termina por llenar la memoria con datos inútiles.

La solución es emplear otra instrucción "ON": ON ERROR GOTO.

En efecto, sólo hay que poner al final de la rutina de interrupción una línea como:

20 RETURN 20

Cuando el programa ha desapilado todas las llamadas, se produce un error del tipo "RETURN without GOSUB". Si con anterioridad se ha previsto una rutina de tratamiento de errores, el programa no se detendrá y la ejecución pasará al tronco principal:

1 ON ERROR GOTO 2
2 RESUME 3
3 ' MENU PRINCIPAL

En fin, espero haber hecho lo suficiente para que te sientas tentado a cuidar más la realización de los programas, sin conformarte con el mero hecho de que funcionen. Al final, todos nos agradecemos haber completado un programa sin "parches", sobre todo después de habernos peleado con algunos "espaguetis".

LISTADOR DE VARIABLES

Una buena forma de pulir la programación es observar como otras personas resuelven sus problemas. Ya que los listados suelen ser largos, hay que empezar por hacer una relación de todas las variables y anotar la función de cada una de ellas. A tal efecto, me ha parecido útil construir una rutina que sirva para eso: listar todas las variables de un programa BASIC, especificando su nombre, su tipo y si son o no de conjunto.

Confío en que te será de alguna ayuda.

A continuación se incluye el listado en ASSEMBLER y el cargador de líneas DATA. Te recomiendo que grabes los bytes de la rutina con BSAVE y así podrás recuperarla con BLOAD. Hay dos puntos a tener en cuenta. El primero es que se debe ejecutar el programa BASIC en su totalidad para que las variables sean definidas. Si modificas una línea, se borrarán todas ellas de la memoria. El segundo punto es que hay que llamar a la rutina con ?USR (0), para salida por pantalla, o ?USR (1), para salida por impresora. En cualquier caso, debes lanzarla con DEFUSER=&HD000.

LISTADO 3

```
10          ORG  #D000
20          LD   (PILA),SP
30          CALL #CC
40          LD   A,12
50          CALL #A2
60          LD   HL,SIM
70          LD   IY,#F6C2
80          LD   E,0
90          CALL PRG
100         INC  IY
110         INC  IY
120         LD   HL,CON
130         LD   E,1
140 PRG:    CALL MES
150         LD   HL,ENT
160         LD   A,2
170         CALL PVAR
180         LD   HL,REAL
190         LD   A,4
200         CALL PVAR
210         LD   HL,DOBLE
220         LD   A,8
230         CALL PVAR
240         LD   HL,CADENA
250         LD   A,3
260         CALL PVAR
270         RET
280 PVAR:
290         CALL MES
300         LD   L,(IY+0)
310         LD   H,(IY+1)
320 LO:     LD   C,(IY+2)
330         LD   B,(IY+3)
340         OR   A
350         SBC  HL,BC
360         ADD  HL,BC
370         RET  Z
380         PUSH HL
390         POP  IX
400         LD   B,0
410         LD   C,(HL)
420         INC  C
430         INC  C
440         INC  C
450         DEC  E
460         INC  E
470         JR   Z,SUMA
480         INC  HL
490         INC  HL
500         INC  HL
510         LD   C,(HL)
520         INC  HL
530         LD   B,(HL)
540         INC  BC
550 SUMA:   ADD  HL,BC
560         CP   (IX+0)
570         JR   NZ,LO
580         LD   A,(IX+1)
590         OR   A
600         RET  Z
610         CALL PRINT
620         LD   A,(IX+2)
630         CALL PRINT
640         LD   A," "
650         CALL PRINT
660         LD   A,(IX+0)
```

```
670          JR   L0
680 MES:
690          PUSH AF
700          CALL LF
710          LD   A,(#F3B0)
720          DEC  A
730          LD   B,A
740 L2:      PUSH BC
750          LD   A,"-"
760          CALL PRINT
770          POP  BC
780          DJNZ L2
790          CALL LF
800 L1:      LD   A,(HL)
810          OR   A
820          JR   Z,EXIT
830          CALL PRINT
840          INC  HL
850          JR   L1
860 EXIT:    POP  AF
870          RET
880 LF:
890          LD   A,13
900          CALL PRINT
910          LD   A,10
920 PRINT:
930          LD   B,A
940          LD   A,(#F7F8)
950          OR   A
960          JR   NZ,LPRINT
970          LD   A,(#F3DC)
980          CP   24
990          JR   NZ,OK
1000         LD   A,(#F3DD)
1010         LD   C,A
1020         LD   A,(#F3B0)
1030         DEC  A
1040         CP   C
1050         JR   Z,KEY
1060         LD   A,B
1070         CP   10
1080         JR   NZ,OK
1090 KEY:    CALL #9F
```



```
1100         CP   3
1110         JR   Z,ERROR
1120 OK:     LD   A,B
1130         JP   #A2
1140 LPRINT: LD   A,B
1150         CALL #A5
1160         RET  NC
1170 ERROR:  LD   SP,(PILA)
1180         RET
1190 SIM:
1200         DEFM "  *** VARIABLES
SIMPLES ***"
1210         DEFB 0
1220 ENT:    DEFB 10,9
1230         DEFM "Enteras"
1240         DEFB 10,10,13,0
1250 REAL:   DEFB 10,9
1260         DEFM "Reales"
1270         DEFB 10,10,13,0
1280 DOBLE:  DEFB 10,9
1290         DEFM "Doble precisi"n"
1300         DEFB 10,10,13,0
1310 CADENA: DEFB 10,9
1320         DEFM "Alfanumricas"
1330         DEFB 10,10,13,0
1340 CON:    DEFB 10,10
1350         DEFM "*** VARIABLES DE
CONJUNTO***"
1360         DEFB 0
1370 PILA:   DEFW 0
```

LISTADO 4

```
10 FORX=&HD000TO&HD165
20 READ V$:POKEX,VAL("&H"+V$)
30 S=S+PEEK(X):NEXT
40 IFS<>36080!THENBEEP:CLS:PRINT"HAY UN
 ERROR"
50 DATAED,73,64,D1,CD,CC,00,3E,0C,CD,A2
,00,21,E6,D0,FD,21,C2,F6,1E,00,CD,21,D0
,FD,23,FD,23,21,45,D1,1E,01,CD,8B,D0,21
,04,D1,3E,02,CD,45,D0,21,11,D1,3E,04,CD
,45,D0,21,1D,D1,3E,08,CD,45,D0,21,32,D1
,3E,03
60 DATACD,45,D0,C9,CD,8B,D0,FD,6E,00,FD
,66,01,FD,4E,02,FD,46,03,B7,ED,42,09,C8
,E5,DD,E1,06,00,4E,0C,0C,0C,1D,1C,28,07
,23,23,23,4E,23,46,03,09,DD,BE,00,20,DB
,DD,7E,01,B7,C8,CD,B3,D0,DD,7E,02,CD,B3
,D0,3E
70 DATA20,CD,B3,D0,DD,7E,00,18,C3,F5,CD
,AC,D0,3A,B0,F3,3D,47,C5,3E,20,CD,B3,D0
,C1,10,F7,CD,AC,D0,7E,B7,28,06,CD,B3,D0
,23,18,F6,F1,C9,3E,0D,CD,B3,D0,3E,0A,47
,3A,F8,F7,B7,20,22,3A,DC,F3,FE,18,20,17
,3A,DD
80 DATAF3,4F,3A,B0,F3,3D,B9,28,05,78,FE
,0A,20,07,CD,9F,00,FE,03,28,09,78,C3,A2
,00,78,CD,A5,00,D0,ED,7B,64,D1,C9,20,20
,20,20,2A,2A,2A,20,56,41,52,49,41,42,4C
,45,53,20,53,49,4D,50,4C,45,53,20,2A,2A
,2A,00
90 DATA0A,09,45,6E,74,65,72,61,73,0A,0A
,0D,00,0A,09,52,65,61,6C,65,73,0A,0A,0D
,00,0A,09,44,6F,62,6C,65,20,70,72,65,63
,69,73,69,A2,6E,0A,0A,0D,00,0A,09,41,6C
,66,61,6E,75,6D,82,72,69,63,61,73,0A,0A
,0D,00
100 DATA0A,0A,2A,2A,2A,20,56,41,52,49,4
1,42,4C,45,53,20,44,45,20,43,4F,4E,4A,5
5,4E,54,4F,2A,2A,2A,00,00,00
```

POR JOAQUIN LOPEZ



I. V. A.

Programa de gestión realizado por Miguel Soler Campos

Se acabaron los quebraderos de cabeza para poder calcular el IVA en vuestras transacciones. Este programa os calcula el IVA de vuestras facturas (clientes y proveedores) y os realiza un balance de fin de año.

Variables:

Principales:

x n.º de apunte

n tipo de factura (1 = clientes, 2 = proveedores)

c$ (x, n) nombre cliente o proveedor

fra (x, n) importe neto facturas

iva (x, n) iva generado

Secundarias:

nf$ leyenda tipo factura

a$ Factura neta o incluyendo iva

I % para cálculo iva

sf (n) suma facturas

si (n) suma iva

fb importe bruto facturas

d día

m mes

ml y m2 auxiliares mes

y, t, a auxiliares para bucles y variables línea

Ampliaciones o mejoras:

Ampliación n.º de facturas

Ordenamiento por fechas, nombre o totales

Impresión (No dispongo de impresora por lo que no he podido añadir las líneas de programa para imprimir)

Dado que el programa lo he realizado para mi uso he hecho las fórmulas de cálculo del IVA repercutido partiendo del 12% que es el que debo de aplicar. Caso de que el % sea otro simplemente se debe variar la línea 1560 cambiando el valor de la variable I.

En caso de que el % de IVA repercutido sea variable deben suprimirse las líneas 1690 a 1710 (inc.) y copiar las líneas 1910 a 2000 adaptando la numeración de acuerdo con las líneas suprimidas.

(Estas dos últimas instrucciones ya están incluidas en el programa)

```
5  'IVA
6 .'POR M.SOLER
7  'PARA MSX EXTRA
10 GOSUB3920
20 MAXFILES=2
30 CLEAR 10000
40 DIMF$(125,2),C$(125,2),FRA(125,2),IVA(125,2),SF(2),SI(2)
50 GOTO1020
60 ' *** Subr. pulsado tecla ***
70 K$=INKEY$:IFK$=""THEN70
80 RETURN
90 '
100 '*** Subrut.imposicion fecha"
110 LOCATE17,5:PRINT"       ":LOCATE1,5:INPUT"Fecha (ddmmaa)";F$(X,N)
120 IF LEN(F$(X,N))<>6THEN110
130 D=VAL(MID$(F$(X,N),1,2))
140 M=VAL(MID$(F$(X,N),3,2))
150 IFM<1ORM>12THEN110
160 IFM=2ORM=4ORM=6ORM=9ORM=11THEN180ELSE170
170 IFD<1ORD>31THEN110ELSE210
180 IFM=2THEN190ELSE200
190 IFD<1ORD>28THEN110ELSE210
200 IFD<1ORD>30THEN110ELSE210
210 RETURN
240 '
250 ' *** Subrut.leyenda ***
260 IF N=1THEN270ELSE280
270 CLS:N$="Facturas clientes":LOCATE12,0:PRINTN$:LOCATE12,1:PRINT"=================":GOTO290
280 CLS:N$="Facturas proveedores":LOCATE10,0:PRINTN$:LOCATE10,1:PRINT"===================="
290 RETURN
```

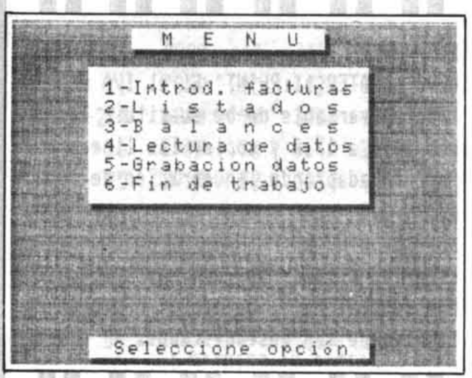
```
300 '
310 '*** Subr. menus secundarios ***
320 LOCATE10,5:PRINT"1-C l i e n t e s"
330 LOCATE10,7:PRINT"2-P r o v e e d o r e s"
340 RETURN
350 LOCATE10,11:PRINT"4-B a l a n c e s"
360 LOCATE10,13:PRINT"5-M e n u"
370 LOCATE11,22:PRINT"Seleccione opcion"
380 RETURN
390 '
400 '*** Subr.introduccion meses ***
410 M1=0:M2=0
420 LOCATE1,6:PRINT"De que mes a que mes?"
430 LOCATE1,8:PRINT"En caso de querer solo un mes ponga la  misma cifra en los dos datos"
440 LOCATE1,11:INPUT"Primer mes :";M1
450 LOCATE1,13:INPUT"Ultimo mes :";M2
460 RETURN
500 '
510 '*** Subr.presentacion listados ***
520 CLS:GOSUB390
530 IFN=1THEN540ELSE550
540 CLS:LOCATE8,0:PRINT"Listado facturas clientes":GOTO560
550 CLS:LOCATE6,0:PRINT"Listado facturas proveedores"
560 LOCATE1,2:PRINT"Fecha    Cliente"
570 LOCATE24,2:PRINT"Factura     IVA"
600 '
610 ' *** Subr.listado facturas ***
620 '
630 SF(N)=0:SI(N)=0
640 Y=4:SF(N)=FRA(W,N):SI(N)=IVA(W,N)
650 FORW=1TOX
```

```
660 M=VAL(MID$(F$(W,N),3,2))
670 IFM<M1ORM>M2THEN760ELSE680
680 IFF$(W,N)=""THEN770ELSE690
690 LOCATE0,Y:PRINTF$(W,N)
700 LOCATE8,Y:PRINTC$(W,N)
710 LOCATE23,Y:PRINTUSING"########";FRA(W,N)
720 LOCATE33,Y:PRINTUSING"#######";IVA(W,N)
730 SF(N)=SF(N)+FRA(W,N):SI(N)=SI(N)+IVA(W,N)
740 Y=Y+1:IFY=18THEN750ELSE760
750 GOSUB830
760 NEXT W
770 LOCATE 0,19:PRINT"Totales :":LOCATE23,19:PRINTUSING"########";SF(N):LOCATE33,19:PRINTUSING"#######";SI(N)
780 LOCATE12,21:PRINT"Pulse una tecla"
790 GOSUB70
795 RETURN
800 '
805 '
810 '
820 ' *** Subr.cambio de pagina ***
830 LOCATE6,21:PRINT"Pulse una tecla par
```

PROGRAMAS

```
a seguir"
840 Y=4:Z=4:GOSUB70
850 FORT=3TO21
860 LOCATE0,T:PRINTSTRING$(40," ")
870 NEXT T
880 RETURN
900 '
910 ' *** Cambio de modo de escrit.***
920 COLOR1,7,7:SCREEN0:WIDTH40:CLS
930 RETURN
1000 '
1010 ' *** Menu principal ***
1020 CLOSE:COLOR1,14.14:SCREEN2:OPEN"grp
:"AS#1:CLS
1030 LINE(72,13)-(183,23),1.BF:LINE(70,1
0)-(180,20),7,BF
1040 LINE(47,33)-(208,103).1,BF:LINE(45.
30)-(205,100),7.BF
1050 LINE(47,173)-(208,183),1,BF:LINE(45
,170)-(205,180),7,BF
1060 DRAW"bm88,12":COLOR1:PRINT#1,"M  E
 N  U"
1070 DRAW"bm55,37":PRINT#1,"1-Introd. fa
cturas"
1080 DRAW"bm55,47":PRINT#1,"2-L i s t a
d o s"
1090 DRAW"bm55,57":PRINT#1,"3-B a l a n
c e s"
1100 DRAW"BM55,67":PRINT#1,"4-Lectura de
 datos"
1110 DRAW"bm55,77":PRINT#1,"5-Grabacion
datos"
1120 DRAW"bm55,87":PRINT#1,"6-Fin de tra
bajo"
1130 DRAW"bm60,172":PRINT#1,"Seleccione
opción"
1140 GOSUB70
1150 K=VAL(K$):IFK<1ORK>7THEN1140
1160 ON K GOTO1500.2200,2420,1200,3410,3
720
1200 '
1210 '*** Lectura datos ***
1220 GOSUB920
1230 LOCATE11,1:PRINT"LECTURA DE DATOS"
1240 LOCATE11,10:PRINT"Pulse PLAY  en el
"
1250 LOCATE11,12:PRINT"cassette y RETURN
"
1255 LOCATE2,21:PRINT"Si quiere volver a
l menu pulse M"
1260 GOSUB70
1265 IFK$="m"ORK$="M"THEN1000
1270 LOCATE11,10:PRINTSTRING$(18," ")
1280 LOCATE11,12:PRINTSTRING$(18," ")
1290 LOCATE13,10:PRINT"Leyendo..."
1300 OPEN"CAS:"FORINPUTAS#2
1310 INPUT#2,X
1320 FORW=1TOX
1330 N=1
1340 INPUT#2,F$(W,N)
1350 INPUT#2,C$(W,N)
1360 INPUT#2,IVA(W,N)
1370 INPUT#2,FRA(W,N)
1380 N=2
1390 INPUT#2,F$(W,N)
1400 INPUT#2,C$(W,N)
1410 INPUT#2,IVA(W,N)
1420 INPUT#2,FRA(W,N)
1430 NEXT W
1440 CLOSE#2
1450 LOCATE11,10:PRINT"Vuelvo al menu."
1460 FORT=1TO1000:NEXT:GOTO1020
1500 '
1510 ' *** Introd. facturas ***
1520 GOSUB920
1530 LOCATE12,1:PRINT"Introd.Facturas"
1540 LOCATE12,2:PRINT"---------------"
1550 GOSUB310:LOCATE10,9:PRINT"3-L i s t
 a d o s":GOSUB350
1560 GOSUB70
1570 K=VAL(K$):IFK<1ORK>5THEN1560
1580 ON KGOTO1590,1810,2200,2420,1020
1590 '*** Facturas clientes ***
1600 X=X+1:N=1
1610 IFX>125THEN3820
1620 GOSUB240
1630 LOCATE1,3:PRINT"Apunte nº:";X
1640 GOSUB90
1660 LOCATE1,7:INPUT"Cliente";C$(X,N)
1670 IFLEN(C$(X,N))>15THEN1680ELSE1690
1680 LOCATE8,7:PRINTSTRING$(30," "):GOTO
1660
1690 LOCATE1,10:INPUT"Importe neto";FRA(
X,N)
1695 I=12
1700 IVA(X,N)=INT(FRA(X,N)*I/100)
1710 LOCATE1,12:PRINT"I V A :";IVA(X,N)
1720 LOCATE10,22:PRINT"Otra factura (s/n
)?"
1730 GOSUB70
1740 IF K$="N"ORK$="n"THEN1510ELSE1750
1750 GOTO1590
1760 GOTO1510
1800 '
1810 '*** Facturas proveedores ***
1820 X=X+1:N=2
1830 IFX>125THEN3820
1840 GOSUB240
1850 LOCATE1,3:PRINT"Apunte nº:";X
1860 GOSUB90
1880 LOCATE1,7:INPUT"Proveedor";C$(X,N)
1890 IFLEN(C$(X,N))>15THEN1900ELSE1910
1900 LOCATE8,7:PRINTSTRING$(30," "):GOTO
1880
1910 LOCATE1,9:PRINT"Factura neta o incl
uye IVA ?"
1920 LOCATE1,11:PRINT"Si es neta pulse N
,si incluye IVA pulse I (y RETURN)"
1930 LOCATE1,13:INPUT"¿N o I";A$
1940 IF A$="N"ORA$="n"THEN1970
1950 IFA$="I"ORA$="i"THEN2050
1960 IF A$<>"n"ORA$<>"N"ORA$<>"I"ORA$<>"
i"THEN1910
1970 LOCATE1,15:INPUT"Importe neto";FRA(
X,N)
1975 I=0
1980 LOCATE1,17:INPUT"% IVA";I
1990 IVA(X,N)=INT(FRA(X,N)*I/100)
2000 LOCATE1,19:PRINT"Importe IVA:";IVA(
X,N)
2010 LOCATE10,22:PRINT"Otra factura (s/n
)?"
2020 GOSUB70
2030 IF K$="N"ORK$="n"THEN1510ELSE1810
2040 GOTO1810
2050 LOCATE1,15:INPUT"Importe bruto";FB
2060 LOCATE1,17:INPUT"% IVA";I
2070 IVA(X,N)=INT((FB*I)/(100+I))
2080 FRA(X,N)=FB-IVA(X,N)
2090 LOCATE1,19:PRINT"Importe IVA:";IVA(
X,N)
2100 LOCATE1,20:PRINT"Importe neto fra."
;FRA(X,N)
2110 LOCATE10,22:PRINT"Otra factura (s/n
)?"
2120 GOSUB70
2130 IF K$="N"ORK$="n"THEN1510ELSE1810
2200 '
2210 ' *** Listados ***
2220 GOSUB920
2230 LOCATE12,1:PRINT"L i s t a d o s"
2240 GOSUB310:LOCATE10,9:PRINT"3-B a l a
 n c e s":LOCATE10,11:PRINT"4-M e n u"
2250 GOSUB70
2260 K=VAL(K$):IFK<1ORK>4THEN2250
2270 ON K GOTO2280,2310,2420,1020
2280 N=1
2290 GOSUB500
2300 GOTO2210
```






```
2310 N=2
2320 GOSUB500
2330 GOTO2210
2400 '
2410 '*** B a l a n c e s ***
2420 CLS:GOSUB910
2430 LOCATE12,1:PRINT"B a l a n c e s"
2440 LOCATE12,2:PRINT"---------------"
2450 LOCATE11,5:PRINT"1-Balance facturas"
2460 LOCATE11,7:PRINT"2-Balance I V A"
2470 LOCATE11,9:PRINT"3-M e n u"
2480 LOCATE10,21:PRINT"Seleccione opcion"
2490 GOSUB70
2500 K=VAL(K$):IFK<1ORK>3THEN2490
2510 ON K GOTO2600,2900,1010
2600 '
2610 '*** Balance facturas ***
2620 CLS:GOSUB400
2630 CLS:LOCATE11,1:PRINT"Balance facturas"
2640 LOCATE11,2:PRINT"----------------"
2650 GOSUB3300
2660 SF(1)=0:SF(2)=0
2670 SF(1)=FRA(W,1):SF(2)=FRA(W,2)
2680 FORN=1TO2
2690 FORW=1TOX
2700 M=VAL(MID$(F$(W,N),3,2))
2710 IFM<M1ORM>M2THEN2750ELSE2720
2720 IFN=1THENIFF$(W,N)=""THEN2760ELSE2740
2730 IFN=2THENIFF$(W,N)=""THEN2770ELSE2740
2740 SF(N)=SF(N)+FRA(W,N)
2750 NEXT W
2760 NEXT N
2770 GOSUB3220
2780 LOCATE4,6:PRINT"Clientes    Proveedores     Saldo"
2790 LOCATE4,9:PRINTUSING"########";SF(1)
2800 LOCATE17,9:PRINTUSING"########";SF(2)
2810 LOCATE29,9:PRINTUSING"########";SF(1)-SF(2)
2820 LOCATE2,21:PRINT"Pulse una tecla para volver al menu"
2830 GOSUB70
2840 GOTO2420
2900 '
2910 '*** Balance I V A ***
2920 GOSUB910
2930 CLS:GOSUB400
2940 CLS:LOCATE12,1:PRINT"Balance I V A"
2950 LOCATE12,2:PRINT"-------------"
2960 GOSUB3300
2970 SI(1)=0:SI(2)=0
2980 SI(1)=IVA(W,1):SI(2)=IVA(W,2)
2990 FORN=1TO2
3000 FORW=1TOX
3010 M=VAL(MID$(F$(W,N),3,2))
3020 IFM<M1ORM>M2THEN3060ELSE3030
3030 IFN=1THENIFF$(W,N)=""THEN3070ELSE3040
3040 IFN=2THENIFF$(W,N)=""THEN3080ELSE3050
3050 SI(N)=SI(N)+IVA(W,N)
3060 NEXT W
3070 NEXT N
3080 GOSUB3220
3090 LOCATE4,6:PRINT"Repercut.    Soportado     Saldo"
3100 LOCATE4,9:PRINTUSING"########";SI(1)
3110 LOCATE17,9:PRINTUSING"########";SI(2)
3120 LOCATE29,9:PRINTUSING"########";SI(1)-SI(2)
3130 LOCATE2,21:PRINT"Pulse una tecla para volver al menu"
3140 GOSUB70
3150 GOTO2420
3200 '
3210 '*** Recuadro titulos balances ***
3220 LOCATE1,5:PRINT"┌------------------WQW------------------┐"
3230 LOCATE1,6:PRINT"|                                     |"
3240 LOCATE1,7:PRINT"├------------------WQW------------------┤"
3250 LOCATE1,8:PRINT"|                                     |"
3260 LOCATE1,9:PRINT"|                                     |"
3270 LOCATE1,10:PRINT"|                                     |"
3280 LOCATE1,11:PRINT"└------------------"QW------------------┘"
3290 RETURN
3300 '
3301 '***
3305 FORY=5TO15:PRINTSTRING$(39," "):NEXT Y
3310 RETURN
3400 '
3410 '*** Grabacion datos ***
3420 GOSUB920
3430 CLS
3440 LOCATE10,1:PRINT"GRABACION DE DATOS"
3450 LOCATE8,10:PRINT"Pulse PLAY y RECORD en"
3460 LOCATE8,12:PRINT"el cassette  y RETURN."
3465 LOCATE2,21:PRINT"Si quiere volver al menu pulse M"
3470 GOSUB70
3475 IFK$="m"ORK$="M"THEN1000
3480 LOCATE8,10:PRINTSTRING$(23," ")
3490 LOCATE8,12:PRINTSTRING$(23," ")
3500 LOCATE10,10:PRINT"    Grabando..."
3510 OPEN"CAS:"FOROUTPUTAS#2
3520 PRINT#2,X
3530 FORW=1TOX
3540 N=1
3550 PRINT#2,F$(W,N)
3560 PRINT#2,C$(W,N)
3570 PRINT#2,IVA(W,N)
3580 PRINT#2,FRA(W,N)
3590 N=2
3600 PRINT#2,F$(W,N)
3610 PRINT#2,C$(W,N)
3620 PRINT#2,IVA(W,N)
3630 PRINT#2,FRA(W,N)
3640 NEXT W
3650 CLOSE#2
3660 LOCATE8,10:PRINTSTRING$(18," ")
3670 LOCATE5,10:PRINT"   Vuelvo al menu."
3680 FORT=1TO1000:NEXT:GOTO1020
3690 GOTO1020
3700 '
3710 '*** Fin de trabajo ***
3720 COLOR15,4,4:CLS:END
3800 '
3810 '*** Subr. exceso de datos ***
```




```
3820 CLS:LOCATE1,10:PRINT"Ya no caben ma
s datos, grábelos.":GOTO1020
3900 '
3910 '### Presentacion ###
3920 COLOR1,14,14:SCREEN2:CLS
3930 CLOSE:OPEN"grp:"AS#1
3940 FORI=1TO8:READ H$:L$=L$+CHR$(VAL("&
B"+H$)):NEXT I
3950 SPRITE$(0)=L$
3960 DATA 00011000,00111100,00111100,111
11111,11111111,00111100,00111100,0001100
0
3970 LINE(25,65)-(231,135),1,BF:LINE(20,
60)-(226,130),7,BF
3980 FORX=100TO140 STEP10
3990 READ Q$
4000 DRAW"bm"+STR$(X)+",71":PRINT#1,Q$
4010 PUTSPRITE0,(X,70),1,0
4020 BEEP
4025 FORI=1TO25:NEXT
4030 NEXT X
4040 PUTSPRITE0,(256,209),1,0
4050 LINE(30,84)-(214,84),1
4060 FORX=30TO210 STEP10
4070 READ Q$
4080 DRAW"bm"+STR$(X)+",91":PRINT#1,Q$
4090 PUTSPRITE0,(X,90),1,0
4100 BEEP
4105 FORI=1TO25:NEXT
4110 NEXT X
4120 PUTSPRITE0,(256,209),1,0
4130 LINE(30,103)-(214,103),1
4140 FORX=100TO140 STEP10
4150 READ Q$
4160 DRAW"bm"+STR$(X)+",110":PRINT#1,Q$
4170 PUTSPRITE0,(X,109),1,0
4180 BEEP
4185 FORI=1TO25:NEXT
4190 NEXT X
4200 PUTSPRITE0,(256,209),1,0
4210 DATA I,,V,,A
4220 DATA M,I,G,U,E,L,,S,O,L,E,R,,C,A,M,
P,O,S
4230 DATA M,,S,,X
4240 FORI=1TO1000:NEXT
4250 KEYOFF:COLOR1,14,14:SCREEN0:WIDTH40
4260 LOCATE13,0:PRINT"Instrucciones"
4270 LOCATE2,2:PRINT"- Este programa le
permite llevar una cuenta corriente de f
acturas clientes y facturas proveedores,
con indicacion del IVA correspondiente."
4280 LOCATE2,7:PRINT"- En caso de que el
 IVA repercutido nosea del 12 % debe var
iarse la linea 1695variando el valor de
I."
4290 LOCATE2,11:PRINT"-Si el IVA repercu
tido es variable de-be sustituir las lin
eas 1690 a 1710 y copiar las lineas 1910
 a 2000 adaptando lanumeración de lineas
."
4300 LOCATE2,16:PRINT"- El programa faci
lita los totales de facturas e IVA,así c
omo los balances, detodo el año o parcia
lmente, de los mesesdeseados."
4310 LOCATE11,21:PRINT"Pulse una tecla"
4320 GOSUB60
4500 CLS:LOCATE2,2:PRINT"- La fecha debe
 introducirse con seis cifras seguidas,
por ej.: 6 de Junio de 1986 debe introdu
cirse como 060686"
4510 LOCATE2,6:PRINT"- El programa no ad
mite fechas ilógi- cas, por ej.:30 de Fe
brero."
4520 LOCATE2,9:PRINT"- Los datos "Client
es" o "Proveedores"no admiten más de 15
caracteres."
4530 LOCATE2,12:PRINT"- El número  maxim
o de facturas es de 125"
4540 LOCATE8,21:PRINT"Pulse RETURN para
comenzar"
4550 GOSUB60:RETURN
```

TEST DE LISTADO

Para utilizar el Test de Listados que ofrecemos al final de cada programa, recordamos que previamente hay que cargar en el ordenador el Programa correspondiente aparecido en nuestro número 10, de octubre pág. 29.

5 - 58	190 -123	420 -165	680 -212	870 -215	1150 - 71	1370 -226	1610 - 63	1850 -136	2060 -194	2330 - 65
6 - 58	200 -125	430 - 89	690 -197	880 -142	1160 - 64	1380 - 80	1620 -139	1860 -245	2070 -206	2400 - 58
7 - 58	210 -142	440 -218	700 -202	900 - 56	1200 - 58	1390 -115	1630 -136	1880 -172	2080 -104	2410 - 58
10 -250	240 - 58	450 -232	710 -196	910 - 58	1210 - 58	1400 -112	1640 -245	1890 -196	2090 -127	2420 - 5
20 -134	250 - 58	460 -142	720 -178	920 - 81	1220 - 54	1410 -233	1660 -186	1900 - 1	2100 -124	2430 - 89
30 -229	260 -211	500 - 58	730 -221	930 -142	1230 - 83	1420 -226	1670 - 10	1910 - 52	2110 -109	2440 -174
40 - 89	270 - 22	510 - 58	740 -141	1000 - 58	1240 - 41	1430 -218	1680 - 36	1920 -239	2120 -225	2450 -160
50 -150	280 -190	520 -251	750 -220	1010 - 58	1250 -253	1440 -234	1690 -255	1930 -131	2130 - 87	2460 - 42
60 - 58	290 -142	530 -241	760 -218	1020 -180	1255 - 39	1450 -119	1695 - 83	1940 - 15	2200 - 58	2470 - 60
70 -223	300 - 58	540 -153	770 - 78	1030 - 13	1260 -225	1460 - 68	1700 -102	1950 - 86	2210 - 58	2480 -210
80 -142	310 - 58	550 -235	780 -168	1040 -213	1265 - 83	1500 - 58	1710 -152	1960 - 95	2220 - 54	2490 -225
90 - 58	320 -215	560 -211	790 -225	1050 -141	1270 -199	1510 - 58	1720 -109	1970 - 4	2230 -131	2500 -142
100 - 58	330 - 49	570 -243	795 -142	1060 -126	1280 -201	1520 - 54	1730 -225	1975 - 73	2240 - 65	2510 -113
110 - 6	340 -142	600 - 58	800 - 58	1070 -150	1290 -162	1530 - 23	1740 - 26	1980 -194	2250 -225	2600 - 58
120 - 69	350 -192	610 - 58	805 - 58	1080 -228	1300 -214	1540 -174	1750 -211	1990 -102	2260 -158	2610 - 58
130 -244	360 - 63	620 - 58	810 - 58	1090 -188	1310 - 63	1550 - 41	1760 -130	2000 -127	2270 -160	2620 - 5
140 -255	370 -212	630 -173	820 - 58	1100 - 25	1320 - 11	1560 -225	1800 - 58	2010 -109	2280 - 79	2630 - 23
150 -125	380 -142	640 - 31	830 -215	1110 - 72	1330 - 79	1570 -235	1810 - 58	2020 -225	2290 -144	2640 -112
160 -162	390 - 58	650 - 11	840 - 16	1120 -114	1340 -115	1580 -208	1820 - 44	2030 - 87	2300 - 65	2650 -139
170 -126	400 - 58	660 -254	850 -214	1130 - 34	1350 -112	1590 - 58	1830 - 63	2040 -176	2310 - 80	2660 - 51
180 -128	410 - 55	670 -108	860 - 15	1140 -225	1360 -233	1600 - 43	1840 -139	2050 - 6	2320 -144	2670 -144

2680 -189	2820 -100	3010 -254	3150 - 20	3310 -142	3510 -160	3650 -234	3930 - 46	4060 -165	4185 -138	4320 -215
2690 - 11	2830 -225	3020 -168	3200 - 58	3400 - 58	3520 - 75	3660 -198	3940 -190	4070 -252	4190 -219	4500 - 59
2700 -254	2840 - 20	3030 -195	3210 - 58	3410 - 58	3530 - 11	3670 -115	3950 -172	4080 -105	4200 -251	4510 -173
2710 - 58	2900 - 58	3040 -217	3220 - 78	3420 - 54	3540 - 79	3680 - 68	3960 -220	4090 -112	4210 - 20	4520 - 21
2720 - 95	2910 - 58	3050 - 88	3230 -234	3430 -159	3550 -127	3690 -150	3970 - 78	4100 -192	4220 -167	4530 -201
2730 -106	2920 - 44	3060 -218	3240 - 70	3440 -200	3560 -124	3700 - 58	3980 -165	4105 -138	4230 - 44	4540 - 60
2740 - 75	2930 - 5	3070 -209	3250 -236	3450 -143	3570 -245	3710 - 58	3990 -252	4110 -219	4240 -105	4550 -215
2750 -218	2940 -159	3080 - 59	3260 -237	3460 -251	3580 -238	3720 -241	4000 -103	4120 -251	4250 -115	
2760 -209	2950 - 40	3090 -101	3270 -236	3465 - 39	3590 - 80	3800 - 58	4010 - 92	4130 -215	4260 -169	
2770 - 59	2960 -139	3100 -120	3280 - 86	3470 -225	3600 -127	3810 - 58	4020 -192	4140 -165	4270 - 57	
2780 -247	2970 - 57	3110 -132	3290 -142	3475 - 83	3610 -124	3820 -187	4025 -138	4150 -252	4280 - 35	
2790 -117	2980 -164	3120 -129	3300 - 58	3480 -203	3620 -245	3900 - 58	4030 -219	4160 -145	4290 -254	
2800 -129	2990 -189	3130 -100	3301 - 58	3490 -205	3630 -238	3910 - 58	4040 -251	4170 -131	4300 -106	TOTAL:
2810 -123	3000 - 11	3140 -225	3305 -146	3500 -237	3640 -218	3920 - 76	4050 -177	4180 -192	4310 -167	48942

TOKENS

Programa de gestión realizado por Miguel Soler Campos

Por fin las instrucciones del BASIC al alcance de vuestros dedos. Con sólo pulsar SHIFT y la inicial de la instrucción correspondiente conseguiréis que ésta aparezca en vuestros programas.

Muchas veces tenemos que teclear programas en los cuales se repiten una serie de instrucciones o funciones, estas son casi siempre las mismas, las que más se utilizan. Hay ordenadores, como es el caso del Spectrum, en los cuales las palabras clave están contenidas en una tecla de tal forma que cuando se pulsa esta aparece sin tener que teclearla; tiene la desventaja de que si en un momento no sabemos donde está una determinada palabra clave podemos tardar bastante tiempo en encontrarla. La solución viene de mano de los ordenadores PC's, en ellos los "tokens" u órdenes se pueden teclear y también se pueden conseguir directamente pulsando la tecla "ALT" con una tecla alfabética.

Con este programa he intentado imitar la técnica de los PC's, como los MSX no disponen de la tecla "ALT", he usado la tecla "SHIFT" para sustituirla, de tal forma que pulsando dicha tecla con cualquier tecla alfabética se obtendrá un "token". La desventaja de este programa es que cuando estemos escribiendo en minúsculas y querramos escribir una mayúscula no podremos utilizar la tecla "SHIFT" si no que tendremos que usar el fijador de mayúsculas.

Los "tokens" que generan cada tecla son los siguientes:

A – ASC(
D – DATA
G – GOTO
J – KEY
M – MID$(
P – PRINT
S – SCREEN
V – VAL
Y – LINE(
B – BEEP
E – ELSE
H – HEX$(
K – KILL"
N – NEXT
Q – PUT SPRITE
T – THEN
W – WIDTH
Z – CIRCLE(
C – CHR$
F – FOR
I – INPUT
L – LOCATE
O – OPEN
R – READ
U – USR
X – XOR
TODAS COMBINADAS CON "SHIFT"

He elegido las palabras que he creído de una utilización más normal, la mayoría se corresponden en la primera letra con la tecla en la que están, de forma que se haga un poco más lógico encontrarla en un momento dado.

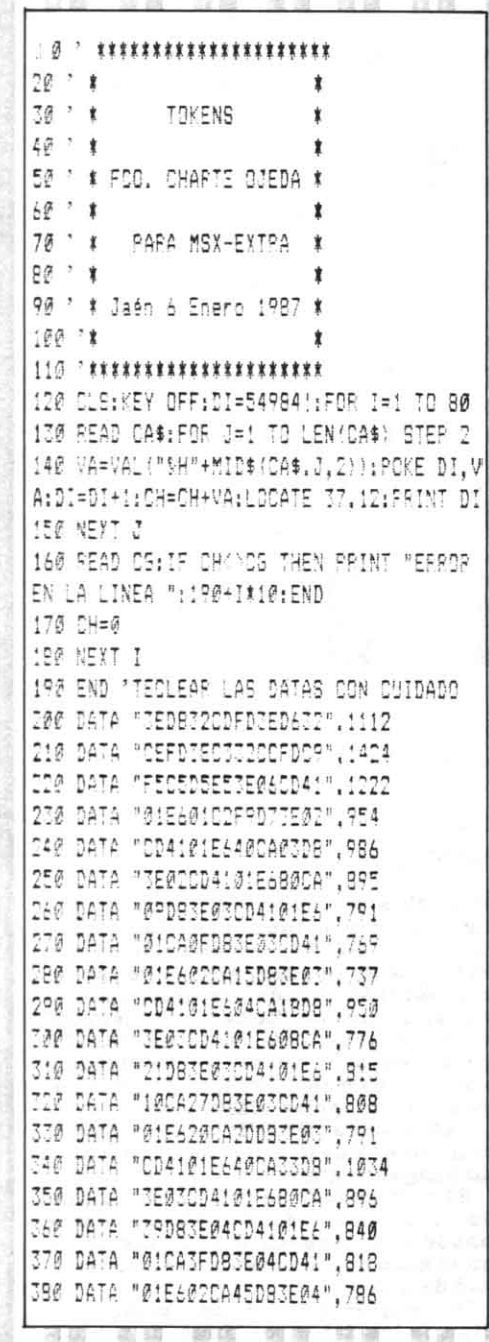
El programa lo que hace es poner un parche en el gancho situado en la dirección &HFDCC, a este gancho se llama cada vez que se pulsa una tecla, de ahí se desvía a la dirección donde está la rutina en asembler, esta lo que hace es inspeccionar si se ha pulsado la tecla "SHIFT", en caso afirmativo averigua si se pulsó alguna de las teclas alfabéticas, de ser así imprime en pantalla el "token" correspondiente a dicha tecla, incrementa el SP para ignorar la última tecla pulsada y vuelve, si no se corresponde con ninguna de las teclas alfabéticas o no se pulsó la tecla "SHIFT" se devuelve el control al BASIC sin haber modificado ningún registro.

El comienzo de la rutina está situado en la dirección 54984, se podría haber aprovechado más la memoria en el caso de los ordenadores sin unidad de disco.

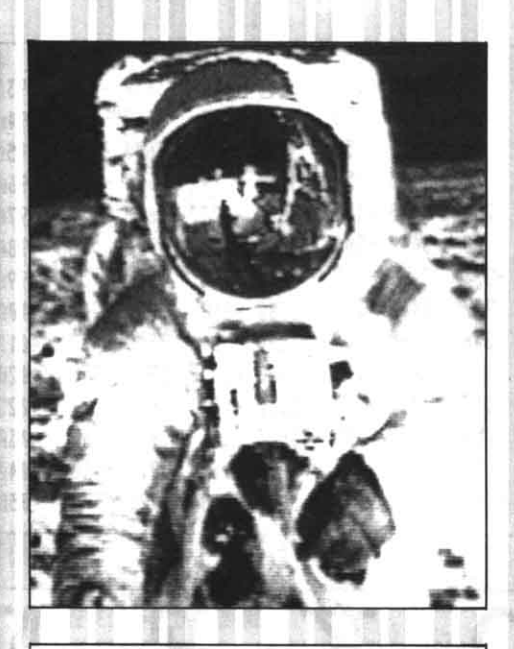
El programa tiene una comprobación de los datos por línea, de tal forma que si hay un error nos diga en qué línea y no tengamos que perder el tiempo mirándolas todas. Una vez que el programa nos haya devuelto el *Ok* tendremos que teclear *DEFUSR=54984: A=USR (0)* y la rutina se habrá activado, a partir de ahora cada vez que pulsemos la tecla "SHIFT" combinada con una alfabética obtendremos el correspondiente "token".

En los ordenadores con unidad de disco si queremos salir del BASIC al MSX-DOS antes de hacerlo deberemos efectuar un POKE en la dirección &HFDCC con el dato 201, si no lo hacemos al salir al MSX-DOS el sistema no funcionará correctamente.

PROGRAMAS



```
10 ' **********************
20 ' *                    *
30 ' *       TOKENS       *
40 ' *                    *
50 ' * FCO. CHARTE OJEDA  *
60 ' *                    *
70 ' *    PARA MSX-EXTRA  *
80 ' *                    *
90 ' * Jaén 6 Enero 1987  *
100 '*                    *
110 '**********************
120 CLS:KEY OFF:DI=54984!:FOR I=1 TO 80
130 READ CA$:FOR J=1 TO LEN(CA$) STEP 2
140 VA=VAL("&H"+MID$(CA$,J,2)):POKE DI,V
A:DI=DI+1:CH=CH+VA:LOCATE 37,12:PRINT DI
150 NEXT J
160 READ CS:IF CH<>CS THEN PRINT "ERROR
EN LA LINEA ";190+I*10:END
170 CH=0
180 NEXT I
190 END 'TECLEAR LAS DATAS CON CUIDADO
200 DATA "3EDB32CDFD3ED632",1112
210 DATA "CEFD3EC332CCFDC9",1424
220 DATA "FEC5D5E53E06CD41",1222
230 DATA "01E601C2F9D73E02",954
240 DATA "CD4101E640CA03D8",986
250 DATA "3E02CD4101E680CA",895
260 DATA "09D83E03CD4101E6",791
270 DATA "01CA0FD83E03CD41",769
280 DATA "01E602CA15D83E03",737
290 DATA "CD4101E604CA1BD8",950
300 DATA "3E03CD4101E608CA",776
310 DATA "21D83E03CD4101E6",815
320 DATA "10CA27D83E03CD41",808
330 DATA "01E620CA2DD83E03",791
340 DATA "CD4101E640CA33D8",1034
350 DATA "3E03CD4101E680CA",896
360 DATA "39D83E04CD4101E6",840
370 DATA "01CA3FD83E04CD41",818
380 DATA "01E602CA45D83E04",786
```



```
390 DATA "CD4101E604CA4BD8",998
400 DATA "3E04CD4101E608CA",777
410 DATA "51D83E04CD4101E6",864
420 DATA "10CA57D83E04CD41",857
430 DATA "01E620CA5DD83E04",840
440 DATA "CD4101E640CA63D8",1082
450 DATA "3E04CD4101E680CA",897
460 DATA "69D83E05CD4101E6",889
470 DATA "01CA6FD83E05CD41",867
480 DATA "01E602CA75D83E05",835
490 DATA "CD4101E604CA7BD8",1046
500 DATA "3E05CD4101E608CA",778
510 DATA "81D83E05CD4101E6",913
520 DATA "10CA87D83E05CD41",906
530 DATA "01E620CA8DD83E05",889
540 DATA "CD4101E640CA93D8",1130
550 DATA "3E05CD4101E680CA",898
560 DATA "99D8C3F9D77EFE23",1443
570 DATA "CAF7D7DF2318F633",1243
580 DATA "333E0132A9FCE1D1",1019
590 DATA "C1F1C9219FD8C3ED",1475
```

```
600 DATA "D721A4D8C3EDD721",1308
610 DATA "A9D8C3EDD721AFD8",1456
620 DATA "C3EDD721B5D8C3ED",1509
630 DATA "D721BBD8C3EDD721",1331
640 DATA "C0D8C3EDD721C6D8",1502
650 DATA "C3EDD721CCD8C3ED",1532
660 DATA "D721D3D8C3EDD721",1355
670 DATA "D8D8C3EDD721DED8",1550
680 DATA "C3EDD721E6D8C3ED",1558
690 DATA "D721ECD8C3EDD721",1380
700 DATA "F2D8C3EDD721F8D8",1602
710 DATA "C3EDD721FFD8C3ED",1583
720 DATA "D7210BD9C3EDD721",1156
730 DATA "11D9C3EDD72119D9",1156
740 DATA "C3EDD7211FD9C3ED",1360
750 DATA "D72123D9C3EDD721",1180
760 DATA "28D9C3EDD7212FD9",1201
770 DATA "C3EDD72134D9C3ED",1381
780 DATA "D7213AD9C3EDD741",1235
790 DATA "5343282342454550",509
800 DATA "2343485224282344",435
810 DATA "4154412023454C53",509
820 DATA "452023464F522023",434
830 DATA "474F544F20234845",521
840 DATA "58242823494E5055",515
850 DATA "5420234B45592023",451
860 DATA "4B494C4C22234C4F",524
870 DATA "434154452023 4D49",502
880 DATA "442428234E455854",498
890 DATA "20234F50454E2223",442
900 DATA "5052494E54202350",544
910 DATA "5554205350524954",603
920 DATA "4520235245414420",452
930 DATA "2353435245454E20",515
940 DATA "235448454E202355",490
950 DATA "535223564l4C2823",502
960 DATA "5749445448202358",539
970 DATA "4F5220234C494E45",524
980 DATA "2823434952434C45",509
990 DATA "2823",75
```

Test de listado

10 - 58	100 - 58	190 -187	280 - 20	370 - 55	460 - 45	550 - 46	640 -125	730 -100	820 -209	910 -204	
20 - 58	110 - 58	200 -122	290 - 49	380 - 28	470 - 63	560 -138	650 -157	740 -142	830 -240	920 -190	
30 - 58	120 - 18	210 -176	300 - 39	390 - 64	480 - 27	570 -116	660 -112	750 - 91	840 -230	930 -215	
40 - 58	130 -136	220 -107	310 - 20	400 - 41	490 -100	580 - 81	670 -153	760 -113	850 -210	940 -222	
50 - 58	140 - 81	230 - 28	320 - 37	410 - 28	500 - 43	590 -137	680 -154	770 -129	860 - 26	950 -213	
60 - 58	150 -205	240 - 42	330 - 36	420 - 45	510 - 27	600 -108	690 -127	780 -109	870 -213	960 -223	
70 - 58	160 -148	250 - 40	340 - 78	430 - 35	520 - 44	610 -154	700 -136	790 -205	880 -238	970 -253	TOTAL:
80 - 58	170 -139	260 - 29	350 - 42	440 - 84	530 - 52	620 -146	710 -169	800 -204	890 -223	980 -227	
90 - 58	180 -204	270 - 56	360 - 28	450 - 44	540 - 81	630 -119	720 -107	810 -212	900 -216	990 - 47	10800

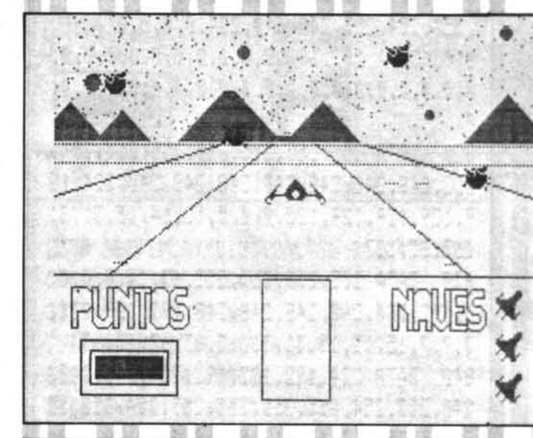


O. V. N. I.

Programa de juego realizado por M.ª Teresa Guillén

Debes evitar que las naves enemigas colisionen con tu nave. Cuentas para ello con tres naves de reserva; pero los enemigos son muchos. Los efectos gráficos están muy bien realizados.

```
5 '****** O.V.N.I. *****
6 '*** por M.T.GUILLEN *
7 '*** para MSX-EXTRA  *
8 '*********************
10 DEFINTA-Z:CLEAR4,57333!
20   COLOR4 ,1,4:SCREEN 2,2
30   LINE(Ø,128)-(255,128)
40   OPEN"GRP:"AS#1
50 '###### PANTALLA 1 #############
60 LINE(Ø,128)-(255,128)
70 FOR X=Ø TO 255 STEP2
80 PSET(X,64):NEXT
90 FOR X=Ø TO 255 STEP 2
100 PSET(X,68):NEXT
110 FOR X=Ø TO 255 STEP 2
120 PSET(X,83):NEXT
130 FOR X=Ø TO 255 STEP 2
140 PSET(X,122):NEXT
150 LINE(140,64)-(220,127)
160 LINE(114,64)-(30,127)
170 LINE(166,64)-(255,90)
180 LINE(88,64)-(Ø,90)
190 REM **TRANSFERIR VRAM1 A RAM1**
200 FOR X=58227!  TO 58239!
210 READV$:POKEX,VAL("&H"+V$)
220 NEXT
230 DATA 1,Ø,8,11,80,E3,21,Ø,8,CD,59,Ø
,C9
240 DEFUSRØ=58227!
250 A=USR(Ø)
260 CLS
270 REM******PANTALLA 2********
280 FOR X=Ø TO 255 STEP 2
290 PSET(X,65):NEXT
300 FOR X=Ø TO 255 STEP 2
310 PSET(X,74):NEXT
320 FOR X=Ø TO 255 STEP 2
330 PSET(X, 97):NEXT
340 LINE(139,65)-(220,127)
350 LINE(115,65)-(30,127)
360 LINE(165,65)-(255,90)
370 LINE(88,65)-(Ø,90)
380 REM*****VRAM2 A RAM2*******
390  FOR X=58227!  TO 58239!
400 READ V$:POKEX,VAL("&H"+V$)
410 NEXT
420 DATA 1,Ø,8,11,80,EB,21,Ø, 8,CD,59,
Ø,C9
430 DEFUSRØ=58227!
440 A=USRØ(Ø)
450 REM *** SUCESION DE IMAGENES HORIZ
ONTALES****************************
460 FOR X=58214!  TO 58226!
470 READV$:POKEX,VAL("&H"+V$)
480 NEXT
490 DATA 1,Ø,8,11,Ø,8,21,80,E3,CD,5C,Ø
,C9
500 DEFUSR=58214!
510 FOR X=58227!  TO 58239!
520 READV$:POKEX,VAL("&H"+V$)
530 NEXT
540 DATA 1,Ø,8,11,Ø,8,21,80,EB,CD,5C,Ø
,C9
550 DEFUSR1=58227!
560 '####### SCROLL ###############
570 FORX=58102! TO 58213!
580  READV$:POKEX,VAL("&H"+V$)
590  NEXT
600 DATA 21,Ø,18,CD,4C,E3,21,F6,DF,DD,
21,F6,DF,6,18,C5,6,1F,7E,F5,23,7E,DD,7
7,Ø,DD,23,10,F7,F1,DD,77,Ø,23,DD,23,C1
,10,E8,CD,59,E3,C9,21,1F,18,CD,4C,E3,2
1,F5,E2,DD,21,F5,E2,6,18
610 DATA C5,6,1F,7E,F5,2B,7E,DD,77,Ø,D
D,2B,10,F7,2B,F1,DD,77,Ø,DD,2B,C1,10,E
8,CD,59,E3,C9,1,Ø,1,21,Ø,18,11,F6,DF,C
D,59,Ø,C9,1,Ø,1,11,Ø,18,21,F6,DF,CD,5C
,Ø,C9
620  DEFUSR2=58102!
630  DEFUSR3=58145!
640  '  ***** FORMA SPRITES  *****
650  FORX=14336 TO 15007
660  READV:VPOKEX,V:NEXT
670  DATA Ø,Ø,Ø,1,2,4,12,24,63,120,144
,128,96,Ø,Ø,Ø,Ø,1,3,7,15,31,30,60,252,
30,15,Ø,Ø,Ø,Ø,Ø
```




```
680  DATA Ø,128,192,224,240,248,120,60
,63,120,240,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,128,64,32,
48,24,252,30,9,1,6,Ø,Ø,Ø
690  DATA Ø,Ø,7,30,127,79,68,32,Ø,Ø,Ø,
Ø,Ø,Ø,Ø,Ø,16,224,131,15,255,63,28,12,1
4,7,3,Ø,Ø,Ø,Ø,Ø
700  DATA Ø,192,224,224,240,248,121,56
,60,60,255,225,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,128
,192,64,96,48,248,124,34,2,4
710  DATA Ø,Ø,Ø,Ø,Ø,Ø,1,3,2,6,12,31,62
,68,64,32,Ø,3,7,7,15,159,30,28,60,60,2
55,135,Ø,Ø,Ø,Ø
720  DATA 8,7,193,240,255,252,56,48,11
2,224,192,Ø,Ø,Ø,Ø,Ø,Ø,Ø,224,120,254,11
4,34,4,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
730  DATA Ø,Ø,70,49,29,45,51,63,63,95,
207,7,Ø,Ø,Ø,Ø,4,24,248,112,128,248,250
,248,248,240,224,200,8,Ø,Ø,Ø
740  DATA Ø,16,1,65,136,17,140,34,25,1
8,69,33,Ø,128,64,Ø,Ø,2,1,170,22,85,170
,239,90,164,66,8,128,Ø,Ø,Ø
750  DATA Ø,32,72,96,217,108,186,85,40
,69,8,8,4,Ø,Ø,Ø,Ø,4,Ø,4,2,37,65,52,169
,80,140,4,17,2,Ø,Ø
760  DATA Ø,Ø,Ø,16,32,193,255,255,127,
63,63,79,135,7,9,16,Ø,Ø,28,60,124,248,
240,224,224,192,192,192,200,208,224,22
4
770  DATA 255,255,255,255,254,252,248,
240,240,240,248,248,252,254,255,255,25
```

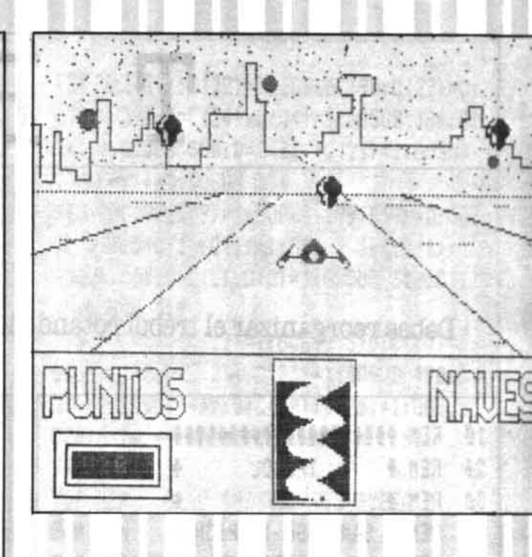
PROGRAMAS

```
4,248,192,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,192,25
3
780  DATA Ø,Ø,Ø,Ø,1,3,15,63,15,3,1,Ø,Ø
,Ø,Ø,Ø,3,7,31,127,255,255,255,255,255,
255,255,127,63,15,7,7
79Ø  DATA 2,4,9,27,19,19,25,12,4,4,6,2
,Ø,2,Ø,Ø,224,240,248,252,252,252,252,2
48,24Ø,24Ø,24Ø,224,192,Ø,Ø,Ø
8ØØ  DATA 1,6,12,25,51,99,1Ø3,1Ø3,63,1
5,15,7,Ø,Ø,Ø,Ø,128,224,24Ø,248,252,254
,254,254,252,24Ø,24Ø,224,Ø,Ø,Ø,Ø
81Ø  DATA 192,192,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø.
Ø
82Ø  DATA 255,255,255,255,255,255,254,
252,248,248,248,248,248,248,24Ø,224,12
9,129,129,129,128,Ø,Ø,Ø,12,14,15,31,31
,63,63,127
83Ø  DATA 255,255,255,255,62,14,Ø,Ø,Ø,
Ø,192,24Ø,248,248,248,248,192,129,131,
3,7,7,15,15,15,31,31,63,63,63,63,63
84Ø  DATA 224,192,192,192,224,24Ø,24Ø,
248,252,254,255,255,255,255,254,252,12
7,255,255,63,63,31,15,6,Ø,Ø,Ø,128,129,
1,1,1
85Ø  DATA 248,249,249,224,192,128,128,
7,15,15,31,63,255,255,255,255,63,63,63
,63,63,127,255,255,255,255,255,255,255
,255,255,255
86Ø  DATA 252,252,252,254,254,254,255,
255,255,255,255,255,255,255,255,255,1,
1,Ø,Ø,Ø,Ø,128,128,24Ø,255,255,255,252,
248,224,192
87Ø  DATA 255,255,255,255,63,47,3,3,3,
3,3,Ø,Ø,Ø,Ø,Ø,255,255,255,255,255,255,
255,255,255,255,255,255,255,255,255,25
5
88Ø '####DIBUJO AFECTADO SCROLL#####
89Ø FORY=ØTO63
9ØØ FORP=125 TO 5ØØ STEP 5Ø
91Ø X=P*RND(TIME)
92Ø PSET(X,Y):NEXT:NEXT
93Ø CIRCLE(1ØØ,2Ø),3 :PAINT(1ØØ,2Ø)
94Ø CIRCLE(199,5Ø),2:PAINT(199,5Ø)
95Ø CIRCLE(2Ø ,35),4:PAINT(2Ø,35)
96Ø CIRCLE(24Ø,1Ø),3:PAINT(24Ø,1Ø)
97Ø LINE(Ø,128)-(255,128)
98Ø PSET(Ø,53)
99Ø DRAW"E9F16E12F2ØR4E3ØR5F22R8E17F4Ø
R15E44F14"
1ØØØ LINE(Ø,64)-(255,64):PAINT(Ø,62):P
AINT(128,61):PAINT(255,61)
1Ø1Ø '#### PANTALLA INFORMACION ####
1Ø2Ø PSET(2Ø,137)
1Ø3Ø DRAW"D6G3L5D8L3U2ØR8F3U3R3D14F2R1
E2U14R3D17G3L5H3U17BR22D2ØL3U8H5D13L3U
2ØR3F5U5R12D3L3D17L3U17L3U3BR19BD3D14G
3L4H3U14E3R4F3E3R6D4L5G1D2F1R3F3D6G3L5
H2U4F2R4E1U2H1L3H3U3"
1Ø4Ø PSET(19Ø,134)
1Ø5Ø DRAW"D2ØL3U8H5D13L3U2ØR3F5U5R3D3E
3R4F3D17L3U5L4D5L3U2ØBR1ØR3D14F2E2U14R
3D16G4L2H4U16BR1ØR9D3L6D5R4D3L4D6R6D3L
9U2ØBR9BD3E3R6D4L5G1D2F1R3F3D6G3L5H2U4
F2R4E1U2H1L3H3U5"
1Ø6Ø  LINE (11Ø,13Ø)-(146,188),4,B
1Ø7Ø PUTSPRITE12,(235,132),15,9
1Ø8Ø PUTSPRITE13,(235,152),15,9
1Ø9Ø PUTSPRITE14,(235,172),15,9
11ØØ LINE(19,165)-(6Ø,181),15,B
111Ø LINE(14,16Ø)-(65,186),15,B
112Ø LINE(22,168)-(57,178),15,BF:PRESE
T(17,17Ø):PRINT#1,P
113Ø '############  ETAPA 1  #######
114Ø I=6:G=1
115Ø ONINTERVAL=3ØØ GOSUB 139Ø
116Ø INTERVAL ON
117Ø PUTSPRITEØ,(112,8Ø)
118Ø PUTSPRITE1,(128,8Ø)
119Ø SOUND5,15:SOUND7,5Ø:SOUND9,16:SOU
ND1Ø,16:SOUND11,5Ø:SOUND12,Ø:SOUND13,1
Ø
12ØØ ON SPRITE GOSUB 2Ø7Ø
121Ø Y= Ø:W=25:T=97:R= 45
122Ø PUTSPRITE8,(X,Y),15,6
123Ø PUTSPRITE9,(X+64 ,W),15,6
124Ø PUTSPRITE1Ø,(X+15Ø,T),15,6
125Ø PUTSPRITE11,(X+192,R),15,6
126Ø SPRITE ON
127Ø A=USR(Ø)
128Ø K=STICK(Ø)
1285 FOR QQ=1 TO 4Ø:NEXT
129Ø IFK=3THENA=USR2(Ø):X=X-2Ø:VPOKE69
14,8:VPOKE6918,12:GOTO132Ø
13ØØ IFK=7THENA=USR3(Ø):X=X+8:VPOKE691
4,16:VPOKE6918,2Ø:GOTO132Ø
131Ø VPOKE6914,Ø:VPOKE6918,4
132Ø IF Y>1Ø8 THEN Y= Ø:X=X-1
133Ø IF W>1Ø8 THEN W=Ø :X=X-2
134Ø IF T>1Ø8 THEN T= Ø:X=X-3
135Ø IF R>1Ø8 THEN R= Ø:X=X-4
136Ø Y=Y+I:W=W+I:T=T+I:R=R+I
137Ø X=X+7
138Ø A=USR1(Ø):FOR QQ=1 TO 4Ø:NEXT:GOT
O 122Ø
139Ø LINE(22,168)-(57,178),15,BF:PRESE
T(17,17Ø):P=P+2Ø :PRINT#1,P
14ØØ I=I+1:IFI<12 THEN RETURN
141Ø '############TRANSICION 1 ####
142Ø GOSUB 229Ø
143Ø '#####  ETAPA 2   ############
144Ø INTERVAL OFF:SPRITE OFF:G=2
145Ø ON INTERVAL=3ØØ GOSUB 158Ø
146Ø PUTSPRITE15,(112,17Ø),7,1Ø
147Ø PUTSPRITE16,(128,17Ø),7,11
148Ø PUTSPRITE17,(112,154),7,1Ø
149Ø PUTSPRITE18,(128,154),7,11
15ØØ PUTSPRITE19,(112,138),7,1Ø
151Ø PUTSPRITE2Ø,(128,138),7,11
152Ø SOUND5,15:SOUND7,5Ø:SOUND9,16:SOU
ND1Ø,16:SOUND11,5Ø:SOUND12,Ø:SOUND13,1
Ø
153Ø X=Ø:Z=131
154Ø Y=Ø:W=25:T=94:R=186
155Ø ON SPRITE GOSUB 2Ø7Ø
156Ø GOSUB189Ø
157Ø A=USR1(Ø):GOTO 156Ø
158Ø LINE(22,168)-(57,178),15,BF:PRESE
T(17,17Ø):P=P+2Ø :PRINT#1,P
159Ø IF R>=136THEN 156Ø
16ØØ '*********** TRANSICION  2  ***
161Ø GOSUB 229Ø
162Ø '############ ETAPA 3 #########
163Ø VPOKE 6974,76
164Ø VPOKE 6978,8Ø
165Ø VPOKE 6982,68
166Ø VPOKE 6986,72
167Ø VPOKE 699Ø,6Ø
168Ø VPOKE 6994,64
169Ø INTERVAL OFF:SPRITE OFF:G=3
17ØØ ON INTERVAL =3ØØ GOSUB 177Ø
171Ø SOUND5,15:SOUND7,5Ø:SOUND9,16:SOU
ND1Ø,16:SOUND11,5Ø:USING12,Ø:SOUND13,1
Ø
172Ø X=Ø:Z=131
173Ø Y=Ø:W=25:T=94:R=186
174Ø ON SPRITE GOSUB 2Ø7Ø
175Ø GOSUB 189Ø
176Ø A=USR1(Ø):GOTO 175Ø
177Ø LINE(22,168)-(57,178),15,BF:PRESE
T(17,17Ø):P=P+2Ø :PRINT#1,P
178Ø IF R>=136THEN 175Ø
18ØØ '####  FINAL ################
181Ø BEEP:SCREEN Ø:KEY OFF:LOCATE 5,Ø
182Ø PRINT"      PUNTUACION:"
183Ø LOCATE22,Ø:PRINT P :END
188Ø '#####RUTINA REPETICION ######
189Ø PUTSPRITE8,(X,Y),15,12
19ØØ PUTSPRITE9,(X+7Ø,W),15,12
```

PROGRAMAS

```
1910 PUTSPRITE10,(X+145,Y),15,12
1920 PUTSPRITE11,(Z,R), 6,14
1930 SPRITE ON:INTERVAL ON
1940 A=USR(0)
1950 K=STICK(0)
1960 IFK=3 THEN A=USR2(0):X=X-8:Z=Z+.2
5:VPOKE6914,8:VPOKE6918,12:GOTO2000
1970 IFK=7 THENA=USR3(0):X=X+8:Z=Z-.25
:VPOKE6914,16:VPOKE6918,20:GOTO2000
1980 VPOKE6914,0:VPOKE6918,4
1990 VPOKE6946,52:VPOKE6950,52:VPOKE69
54,52
2000 IF Y>111THEN Y=0
2010 IF W>111THEN W=0
2020 IF T>111THEN T=0
2030 X=X+4
2040 Y=Y+8:W=W+8:T=T+8:R=R-.1
2050 RETURN
2060 '###### COLISION #############
2070 SPRITE OFF
2080 VPOKE 6914,124:VPOKE6918,124
2090 SOUND 7,0:SOUND 6,30:SOUND9,16:SO
UND10,16:SOUND12,56:SOUND13,0
2100 PUTSPRITE 6,(120,80),15,7
2110 PUTSPRITE 7,(120,80),15,8
```

```
2120 FOR R=0 TO 100:NEXT
2130 PUTSPRITE 6,(116,80),15,7
2140 PUTSPRITE 7,(124,80),15,8
2150 FOR R=0 TO 100:NEXT
2160 PUTSPRITE 6,(112,80),15,7
2170 PUTSPRITE 7,(128,80),15,8
2180 FOR R=0 TO 100:NEXT
2190 PUTSPRITE 6,(128,80),15,7
2200 PUTSPRITE 7,(112,80),15,8
2210 FOR R=0 TO 300:NEXT
2220 VPOKE 6936,191:VPOKE 6940,191
2230 S=S+1:IFS=4THEN2270
2240 VPOKE6959+(4*S),6
2250 PUTSPRITE1,(128,80!)
2260 X=0:ON G GOTO 1170,1520,1710
2270 GOTO 1800
2290 '*** TRANSICION  ETAPAS ******
2300 PAINT(0,62),1:PAINT(128,61),1:PAI
NT(255,61),1
2310 PSET(0,60):DRAW"R5U20R3D15R7D4R6U
15R3U17R6D20R10U4R4U5R3U6R2D6R4D7R2D8R
15U7R4U14R16U20R4D15R4D20R25U8R7U8R6U7
L4H2U4E2R12F2D4G2L4D20R40U4R5U7R3U5R6D
8R4D7R10D5R4D4R6U14R7U3R5D10R5D3R5U12R
3U7R4D8R14D6R4D7R4D8R7U14R4"
```




```
2320 BEEP:PLAY"V15T120L4S8M85004DED5ED
O4DEFC5DCO4DCCO3CABC"
2330 FOR F=1 TO 80
2340 A=USR(0):COLOR 4,1,4
2350 A=USR2(0):COLOR 1,4,1
2360 A=USR1(0):NEXT
2370 COLOR 4,1,4
2380 RETURN
```

TEST DE LISTADO

5 - 58	190 - 0	410 -131	630 -149	850 - 23	1070 - 91	1285 -242	1500 -230	1720 -109	1990 - 42	2210 -179	
6 - 58	200 - 91	420 -234	640 - 58	860 -185	1080 -112	1290 - 35	1510 -248	1730 - 47	2000 -131	2220 - 92	
7 - 58	210 - 88	430 -192	650 -229	870 -203	1090 -133	1300 - 43	1520 -166	1740 - 21	2010 -127	2230 -130	
8 - 58	220 -131	440 -128	660 -116	880 - 58	1100 - 56	1310 -186	1530 -109	1750 - 4	2020 -121	2240 - 12	
10 - 98	230 -219	450 - 0	670 -167	890 - 2	1110 - 56	1320 - 93	1540 - 47	1760 - 46	2030 -165	2250 -122	
20 -162	240 -192	460 -249	680 -126	900 - 86	1120 - 77	1330 - 90	1550 - 21	1770 - 42	2040 -147	2260 -184	
30 -216	250 -111	470 - 88	690 -185	910 - 47	1130 - 58	1340 - 85	1560 - 4	1780 - 21	2050 -142	2270 -166	
40 -224	260 -159	480 -131	700 -103	920 -106	1140 -209	1350 - 82	1570 -112	1800 - 58	2060 - 58	2290 - 58	
50 - 58	270 - 0	490 -229	710 -148	930 - 27	1150 -137	1360 -254	1580 - 42	1810 - 38	2070 -178	2300 -117	
60 -216	280 -176	500 -126	720 - 21	940 - 28	1160 - 67	1370 -168	1590 - 87	1820 - 21	2080 -170	2310 -173	
70 -176	290 -164	510 - 91	730 -118	950 -154	1170 - 18	1380 - 70	1600 - 58	1830 - 16	2090 - 47	2320 -229	
80 -163	300 -176	520 - 88	740 - 24	960 - 31	1180 -122	1390 - 42	1610 -149	1880 - 58	2100 -174	2330 - 13	
90 -176	310 -173	530 -131	750 -140	970 -216	1190 -166	1400 - 5	1620 - 58	1890 -126	2110 -176	2340 -250	
100 -167	320 -176	540 -244	760 -176	980 -148	1200 - 21	1410 - 58	1630 -194	1900 -195	2120 -221	2350 - 10	
110 -176	330 -196	550 -193	770 -159	990 -204	1210 -165	1420 -149	1640 -202	1910 - 15	2130 -170	2360 - 62	
120 -182	340 -254	560 - 58	780 - 8	1000 - 77	1220 -122	1430 - 58	1650 -194	1920 -117	2140 -180	2370 - 81	
130 -176	350 - 40	570 -119	790 - 98	1010 - 58	1230 -185	1440 - 8	1660 -202	1930 -217	2150 -221	2380 -142	
140 -221	360 - 22	580 - 88	800 - 35	1020 -250	1240 - 11	1450 - 72	1670 -194	1940 -111	2160 -166		
150 -254	370 -204	590 -131	810 -176	1030 -107	1250 - 52	1460 - 2	1680 -202	1950 - 61	2170 -184		
160 - 38	380 - 0	600 - 93	820 -147	1040 -161	1260 - 92	1470 - 20	1690 - 9	1960 -105	2180 -221	TOTAL:	
170 - 22	390 - 91	610 - 44	830 -244	1050 -147	1270 -111	1480 -244	1700 - 6	1970 -110	2190 -182		
180 -203	400 - 88	620 - 64	840 -124	1060 -196	1280 - 61	1490 - 6	1710 -198	1980 -186	2200 -168	28020	

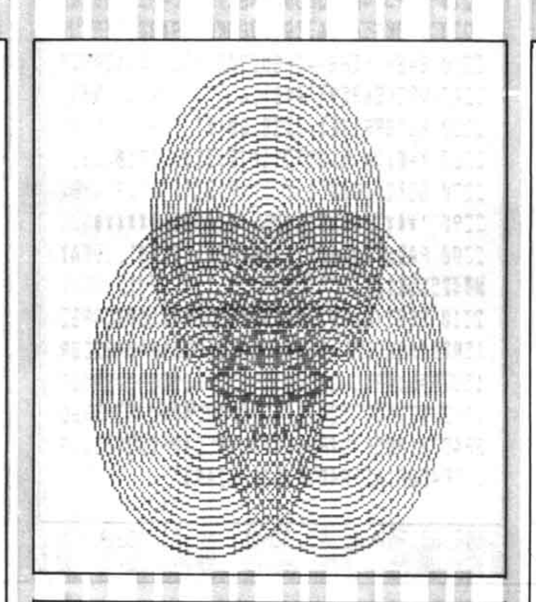


T R E B O L

Programa de juego realizado por
Santiago Turrión

Debes reorganizar el trébol rotando los círculos que lo componen. Un reto para los amantes de los rompecabezas.

```
10  REM ♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣
20  REM ♣      TREBOL       ♣
30  REM ♣        DE         ♣
40  REM ♣ SANTIAGO TURRION  ♣
50  REM ♣       PARA        ♣
60  REM ♣     MSX EXTRA     ♣
70  REM ♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣♣
80 REM
90 REM  PRESENTACION E INSTRUCCIONES
100 REM
110 DEFINT C-Z:FOR I=1 TO 10:KEY I,"":
NEXT:COLOR 4,1,1:SCREEN 2,,0:CLS:R=66:
B=60*ATN(1)/45:B2=2*B:B3=3*B:B4=4*B:B5
=5*B:A=255/191:FOR I=1 TO 66 STEP 3:CI
RCLE(128,67),R-I,,,,A:CIRCLE(103,124),
R-I,,,,A:CIRCLE(153,124),R-I,,,,A:NEXT
120 SCREEN 3:OPEN"GRP:"AS 1:FOR I=2 TO
 15:COLOR I:PRESET(35,85):PRINT#1,"TRE
BOL":FOR J=1 TO 500:NEXT:NEXT:DIM C(24
)
130 COLOR 4:SCREEN 1:LOCATE 9,6:PRINT"
¿QUE QUIERES?":LOCATE 11,12:PRINT "1-
JUGAR":LOCATE 8,18:PRINT "2- EXPERIMEN
TAR"
140 H$=INKEY$:IF H$="" OR INSTR("12",H
$)=0 THEN 140 ELSE:SCREEN 0:LOCATE 16,
7:PRINT"ELIGE":LOCATE 3,15:PRINT"GRADO
 DE DIFICULTAD ENTRE 1 Y 3"
150 E$=INKEY$:IF E$="" OR INSTR("123",
E$)=0 THEN 150 ELSE FOR I=1 TO 9:C(I)=
15:NEXT:FOR I=10 TO 15:C(I)=4:NEXT:IF
E$="1" THEN FOR I=16 TO 24:C(I)=4:NEXT
 ELSE FOR I=22 TO 24:C(I)=13:NEXT:FOR
I=16 TO 21:IF E$="2" THEN C(I)=4:NEXT
ELSE C(I)=2:NEXT
160 SCREEN 1:LOCATE 0,11:PRINT"¿NECESI
TAS INSTRUCCIONES? S/N"
170 B$=INKEY$:IF B$="" OR INSTR("NnSs"
,B$)=0 THEN 170 ELSE IF INSTR("Nn",B$)
 THEN LOCATE 0,11:PRINT SPC(4);"¡UN MO
MENTO POR FAVOR!";SPC(3):GOTO 240
180 SCREEN 0:PRINT" DEBES SITUAR LAS P
```



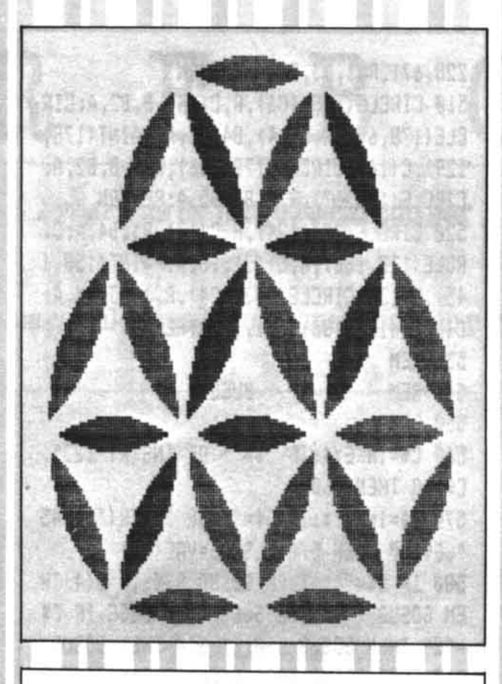
```
IEZAS TAL Y COMO":PRINT:PRINT"APARECEN
 EN LA SIGUIENTE PANTALLA.":PRINT:PRIN
T:PRINT" PARA ELLO PULSA PRIMERO EL NU
MERO DE":PRINT"LA RUEDA Y A CONTINUACI
ON EL NUMERO":PRINT:PRINT"DEL ANGULO D
E GIRO."
190 LOCATE 4,11:PRINT"- EL NUMERO DE L
AS RUEDAS ES:":LOCATE 18,13:PRINT"1":L
OCATE 16,15:PRINT"2   3":LOCATE 4,17:P
RINT"- EL NUMERO DE LOS GIROS ES:":LOC
ATE 14,19:PRINT"1,2,3,4,5":LOCATE 3,21
:PRINT"««« PULSA CUALQUIER TECLA »»»"
200 D$=INKEY$:IF D$="" THEN 200 ELSE I
F H$="2" THEN 240 ELSE SCREEN 2:GOSUB
280
210 REM
220 REM      POSICION ALEATORIA
230 REM
240 IF H$="2" THEN SCREEN 2:GOSUB 280:
GOTO 560 ELSE K=21*RND(-TIME)+20:FOR I
=1 TO K:C=3*RND(-TIME)+1:G=5*RND(-TIME
)+1:ON C GOSUB 620,660,700:NEXT:SCREEN
 2:GOSUB 280:GOTO 560
250 REM
260 REM           PANTALLA
270 REM
280 GOSUB 300:GOSUB 310:GOSUB 410:GOSU
B 420:GOSUB 430:GOSUB 480:GOSUB 490:GO
SUB 500:GOSUB 290:GOSUB 330:GOSUB 360:
GOSUB 370:GOSUB 450:GOSUB 440:GOSUB 32
0:GOSUB 380:GOSUB 390:GOSUB 400:GOSUB
340:GOSUB 350:GOSUB 470:GOSUB 460:GOSU
B 520:GOSUB 510:RETURN
290 CIRCLE(128,67),R,C(9),,B,A:CIRCLE(
203,5),R,C(9),B3,B4,A:PAINT(156,17),C(
9):CIRCLE(128,67),R,1,,B,A:CIRCLE(203,
5),R,1,B3,B4,A:RETURN
300 CIRCLE(128,67),R,C(1),B,B2,A:CIRCL
E(128,-52),R,C(1),B4,B5,A:PAINT(128,2)
,C(1):CIRCLE(128,67),R,1,B,B2,A:CIRCLE
(128,-52),R,1,B4,B5,A:RETURN
310 CIRCLE(128,67),R,C(2),B2,B3,A:CIRC
LE(53,5),R,C(2),B5,,A:PAINT(100,15),C(
2):CIRCLE(128,67),R,1,B2,B3,A:CIRCLE(5
3,5),R,1,B5,,A:RETURN
320 CIRCLE(128,67),R,C(20),B5,,A:CIRCL
E(203,124),R,C(20),B2,B3,A:PAINT(176,7
0),C(20):CIRCLE(128,67),R,1,B5,,A:CIRC
LE(203,124),R,1,B2,B3,A:RETURN
330 CIRCLE(153,124),R,C(21),B,B2,A:CIR
CLE(153,5),R,C(21),B4,B5,A:PAINT(153,6
2),C(21):CIRCLE(153,124),R,1,B,B2,A:CI
RCLE(153,5),R,1,B4,B5,A:RETURN
340 CIRCLE(103,5),R,C(15),B5,,A:CIRCLE
(178,67),R,C(15),B2,B3,A:PAINT(150,15)
,C(15):CIRCLE(103,5),R,1,B5,,A:CIRCLE(
178,67),R,1,B2,B3,A:RETURN
350 CIRCLE(153,5),R,C(10),B3,B4,A:CIRC
LE(78,67),R,C(10),,B,A:PAINT(107,16),C
(10):CIRCLE(153,5),R,1,B3,B4,A:CIRCLE(
78,67),R,1,,B,A:RETURN
360 CIRCLE(103,124),R,C(16),B,B2,A:CIR
CLE(103,5),R,C(16),B4,B5,A:PAINT(120,6
5),C(16):CIRCLE(103,124),R,1,B,B2,A:CI
RCLE(103,5),R,1,B4,B5,A:RETURN
370 CIRCLE(128,67),R,C(17),B3,B4,A:CIR
CLE(53,124),R,C(17),,B,A:PAINT(80,70),
```

PROGRAMAS

```
C(17):CIRCLE(128,67),R,1,B3,B4,A:CIRCL
E(53,124),R,1,,B,A:RETURN
380 CIRCLE(103,124),R,C(24),,B,A:CIRCL
E(178,67),R,C(24),B3,B4,A:PAINT(130,70
),C(24):CIRCLE(103,124),R,1,,B,A:CIRCL
E(178,67),R,1,B3,B4,A:RETURN
390 CIRCLE(153,124),R,C(22),B2,B3,A:CI
RCLE(78,67),R,C(22),B5,,A:PAINT(126,72
),C(22):CIRCLE(153,124),R,1,B2,B3,A:CI
RCLE(78,67),R,1,B5,,A:RETURN
400 CIRCLE(128,67),R,C(23),B4,B5,A:CIR
CLE(128,186),R,C(23),B,B2,A:PAINT(128,
124),C(23):CIRCLE(128,67),R,1,B4,B5,A:
CIRCLE(128,186),R,1,B,B2,A:RETURN
410 CIRCLE(103,124),R,C(3),B2,B3,A:CIR
CLE(28,67),R,C(3),B5,,A:PAINT(75,72),C
(3):CIRCLE(103,124),R,1,B2,B3,A:CIRCLE
(28,67),R,1,B5,,A:RETURN
420 CIRCLE(103,124),R,C(4),B3,B4,A:CIR
CLE(28,186),R,C(4),,B,A:PAINT(56,134),
C(4):CIRCLE(103,124),R,1,B3,B4,A:CIRCL
E(28,186),R,1,,B,A:RETURN
430 CIRCLE(103,124),R,C(5),B4,B5,A:CIR
CLE(103,243),R,C(5),B,B2,A:PAINT(103,1
81),C(5):CIRCLE(103,124),R,1,B4,B5,A:C
IRCLE(103,243),R,1,B,B2,A:RETURN
440 CIRCLE(103,124),R,C(19),B5,,A:CIRC
LE(178,186),R,C(19),B2,B3,A:PAINT(132,
176),C(19):CIRCLE(103,124),R,1,B5,,A:C
IRCLE(178,186),R,1,B2,B3,A:RETURN
450 CIRCLE(153,124),R,C(18),B3,B4,A:CI
RCLE(78,186),R,C(18),,B,A:PAINT(106,14
0),C(18):CIRCLE(153,124),R,1,B3,B4,A:C
IRCLE(78,186),R,1,,B,A:RETURN
460 CIRCLE(53,124),R,C(12),B5,,A:CIRCL
E(128,186),R,C(12),B2,B3,A:PAINT(100,1
33),C(12):CIRCLE(53,124),R,1,B5,,A:CIR
CLE(128,186),R,1,B2,B3,A:RETURN
470 CIRCLE(78,67),R,C(11),B4,B5,A:CIRC
LE(78,186),R,C(11),B,B2,A:PAINT(78,124
),C(11):CIRCLE(78,67),R,1,B4,B5,A:CIRC
LE(78,186),R,1,B,B2,A:RETURN
480 CIRCLE(153,124),R,C(6),B4,B5,A:CIR
CLE(153,243),R,C(6),B,B2,A:PAINT(153,1
81),C(6):CIRCLE(153,124),R,1,B4,B5,A:C
IRCLE(153,243),R,1,B,B2,A:RETURN
490 CIRCLE(153,124),R,C(7),B5,,A:CIRCL
E(228,186),R,C(7),B2,B3,A:PAINT(200,13
8),C(7):CIRCLE(153,124),R,1,B5,,A:CIRC
LE(228,186),R,1,B2,B3,A:RETURN
500 CIRCLE(153,124),R,C(8),,B,A:CIRCLE
(228,67),R,C(8),B3,B4,A:PAINT(200,120)
,C(8):CIRCLE(153,124),R,1,,B,A:CIRCLE(
228,67),R,1,B3,B4,A:RETURN
510 CIRCLE(178,186),R,C(14),B,B2,A:CIR
CLE(178,67),R,C(14),B4,B5,A:PAINT(178,
129),C(14):CIRCLE(178,186),R,1,B,B2,A:
CIRCLE(178,67),R,1,B4,B5,A:RETURN
520 CIRCLE(203,124),R,C(13),B3,B4,A:CI
RCLE(128,186),R,C(13),,B,A:PAINT(158,1
45),C(13):CIRCLE(203,124),R,1,B3,B4,A:
CIRCLE(128,186),R,1,,B,A:RETURN
530 REM
540 REM          BUCLE
550 REM
560 C$=INKEY$:IF C$="" OR INSTR("123",
C$)=0 THEN 560
570 G$=INKEY$:IF G$="" OR INSTR("12345
",G$)=0 THEN 570 ELSE G=VAL(G$)
580 IF C$="1" THEN GOSUB 620:IF G<4 TH
EN GOSUB 740 ELSE GOSUB 950 ELSE IF C$
="2" THEN GOSUB 660:IF G<4 THEN GOSUB
810 ELSE GOSUB 1020 ELSE IF C$="3" THE
N GOSUB 700:IF G<4 THEN GOSUB 880 ELSE
 GOSUB 1090
590 REM
600 REM         GIRO RUEDA 1
610 REM
620 FOR J=1 TO G:D=C(20):C(20)=C(23):C
(23)=C(17):C(17)=C(2):C(2)=C(1):C(1)=C
(9):C(9)=D:D=C(24):C(24)=C(22):C(22)=C
(16):C(16)=C(10):C(10)=C(15):C(15)=C(2
1):C(21)=D:NEXT:RETURN
630 REM
640 REM         GIRO RUEDA 2
650 REM
660 FOR J=1 TO G:D=C(19):C(19)=C(5):C(
5)=C(4):C(4)=C(3):C(3)=C(16):C(16)=C(2
4):C(24)=D:D=C(18):C(18)=C(12):C(12)=C
(11):C(11)=C(17):C(17)=C(22):C(22)=C(2
3):C(23)=D:NEXT:RETURN
670 REM
680 REM         GIRO RUEDA 3
690 REM
700 FOR J=1 TO G:D=C(7):C(7)=C(6):C(6)
=C(18):C(18)=C(22):C(22)=C(21):C(21)=C
(8):C(8)=D:D=C(13):C(13)=C(19):C(19)=C
(23):C(23)=C(24):C(24)=C(20):C(20)=C(1
4):C(14)=D:NEXT:RETURN
710 REM
720 REM   GIRO ANTIHORARIO RUEDA 1
730 REM
740 D=C(15):C(15)=1:GOSUB 340:C(15)=D:
GOSUB 340:D=C(10):C(10)=1:GOSUB 350:C(
10)=D:GOSUB 350:D=C(16):C(16)=1:GOSUB
360:C(16)=D:GOSUB 360
750 D=C(22):C(22)=1:GOSUB 390:C(22)=D:
GOSUB 390:D=C(24):C(24)=1:GOSUB 380:C(
24)=D:GOSUB 380:D=C(21):C(21)=1:GOSUB
330:C(21)=D:GOSUB 330
760 D=C(9):C(9)=1:GOSUB 290:C(9)=D:GOS
UB 290:D=C(1):C(1)=1:GOSUB 300:C(1)=D:
GOSUB 300:D=C(2):C(2)=1:GOSUB 310:C(2)
=D:GOSUB 310
770 D=C(17):C(17)=1:GOSUB 370:C(17)=D:
GOSUB 370:D=C(23):C(23)=1:GOSUB 400:C(
23)=D:GOSUB 400:D=C(20):C(20)=1:GOSUB
320:C(20)=D:GOSUB 320:RETURN 1160
780 REM
790 REM   GIRO ANTIHORARIO RUEDA 2
800 REM
810 D=C(22):C(22)=1:GOSUB 390:C(22)=D:
GOSUB 390:D=C(17):C(17)=1:GOSUB 370:C(
17)=D:GOSUB 370:D=C(11):C(11)=1:GOSUB
470:C(11)=D:GOSUB 470
820 D=C(12):C(12)=1:GOSUB 460:C(12)=D:
GOSUB 460:D=C(18):C(18)=1:GOSUB 450:C(
18)=D:GOSUB 450:D=C(23):C(23)=1:GOSUB
400:C(23)=D:GOSUB 400
830 D=C(24):C(24)=1:GOSUB 380:C(24)=D:
GOSUB 380:D=C(16):C(16)=1:GOSUB 360:C(
16)=D:GOSUB 360:D=C(3):C(3)=1:GOSUB 41
0:C(3)=D:GOSUB 410
840 D=C(4):C(4)=1:GOSUB 420:C(4)=D:GOS
UB 420:D=C(5):C(5)=1:GOSUB 430:C(5)=D:
GOSUB 430:D=C(19):C(19)=1:GOSUB 440:C(
19)=D:GOSUB 440:RETURN 1160
850 REM
860 REM   GIRO ANTIHORARIO RUEDA 3
870 REM
880 D=C(20):C(20)=1:GOSUB 320:C(20)=D:
GOSUB 320:D=C(24):C(24)=1:GOSUB 380:C(
24)=D:GOSUB 380:D=C(23):C(23)=1:GOSUB
400:C(23)=D:GOSUB 400
890 D=C(19):C(19)=1:GOSUB 440:C(19)=D:
GOSUB 440:D=C(13):C(13)=1:GOSUB 520:C(
13)=D:GOSUB 520:D=C(14):C(14)=1:GOSUB
510:C(14)=D:GOSUB 510
900 D=C(8):C(8)=1:GOSUB 500:C(8)=D:GOS
UB 500:D=C(21):C(21)=1:GOSUB 330:C(21)
=D:GOSUB 330:D=C(22):C(22)=1:GOSUB 390
:C(22)=D:GOSUB 390
910 D=C(18):C(18)=1:GOSUB 450:C(18)=D:
GOSUB 450:D=C(6):C(6)=1:GOSUB 480:C(6)
=D:GOSUB 480:D=C(7):C(7)=1:GOSUB 490:C
(7)=D:GOSUB 490:RETURN 1160
920 REM
930 REM     GIRO HORARIO RUEDA 1
940 REM
```

PROGRAMAS

```
950 D=C(24):C(24)=1:GOSUB 380:C(24)=D:
GOSUB 380:D=C(22):C(22)=1:GOSUB 390:C(
22)=D:GOSUB 390:D=C(16):C(16)=1:GOSUB
360:C(16)=D:GOSUB 360
960 D=C(10):C(10)=1:GOSUB 350:C(10)=D:
GOSUB 350:D=C(15):C(15)=1:GOSUB 340:C(
15)=D:GOSUB 340:D=C(21):C(21)=1:GOSUB
330:C(21)=D:GOSUB 330
970 D=C(20):C(20)=1:GOSUB 320:C(20)=D:
GOSUB 320:D=C(23):C(23)=1:GOSUB 400:C(
23)=D:GOSUB 400:D=C(17):C(17)=1:GOSUB
370:C(17)=D:GOSUB 370
980 D=C(2):C(2)=1:GOSUB 310:C(2)=D:GOS
UB 310:D=C(1):C(1)=1:GOSUB 300:C(1)=D:
GOSUB 300:D=C(9):C(9)=1:GOSUB 290:C(9)
=D:GOSUB 290:RETURN 1160
990 REM
1000 REM     GIRO HORARIO RUEDA 2
1010 REM
1020 D=C(18):C(18)=1:GOSUB 450:C(18)=D
:GOSUB 450:D=C(12):C(12)=1:GOSUB 460:C
(12)=D:GOSUB 460:D=C(11):C(11)=1:GOSUB
 470:C(11)=D:GOSUB 470
1030 D=C(17):C(17)=1:GOSUB 370:C(17)=D
:GOSUB 370:D=C(22):C(22)=1:GOSUB 390:C
(22)=D:GOSUB 390:D=C(23):C(23)=1:GOSUB
 400:C(23)=D:GOSUB 400
1040 D=C(19):C(19)=1:GOSUB 440:C(19)=D
:GOSUB 440:D=C(5):C(5)=1:GOSUB 430:C(5
)=D:GOSUB 430:D=C(4):C(4)=1:GOSUB 420:
C(4)=D:GOSUB 420
1050 D=C(3):C(3)=1:GOSUB 410:C(3)=D:GO
SUB 410:D=C(16):C(16)=1:GOSUB 360:C(16
)=D:GOSUB 360:D=C(24):C(24)=1:GOSUB 38
```



```
0:C(24)=D:GOSUB 380:RETURN 1160
1060 REM
1070 REM     GIRO HORARIO RUEDA 3
1080 REM
1090 D=C(13):C(13)=1:GOSUB 520:C(13)=D
:GOSUB 520:D=C(19):C(19)=1:GOSUB 440:C
(19)=D:GOSUB 440:D=C(23):C(23)=1:GOSUB
 400:C(23)=D:GOSUB 400
1100 D=C(24):C(24)=1:GOSUB 380:C(24)=D
:GOSUB 380:D=C(20):C(20)=1:GOSUB 320:C
(20)=D:GOSUB 320:D=C(14):C(14)=1:GOSUB
 510:C(14)=D:GOSUB 510
1110 D=C(7):C(7)=1:GOSUB 490:C(7)=D:GO
SUB 490:D=C(6):C(6)=1:GOSUB 480:C(6)=D
```

```
:GOSUB 480:D=C(18):C(18)=1:GOSUB 450:C
(18)=D:GOSUB 450
1120 D=C(22):C(22)=1:GOSUB 390:C(22)=D
:GOSUB 390:D=C(21):C(21)=1:GOSUB 330:C
(21)=D:GOSUB 330:D=C(8):C(8)=1:GOSUB 5
00:C(8)=D:GOSUB 500:RETURN 1160
1130 REM
1140 REM COMPROBACION DE LA SOLUCION
1150 REM
1160 IF H$="2" THEN 560 ELSE FOR I=1 T
O 9:IF C(I)<>15 THEN 560 ELSE NEXT:FOR
 I=10 TO 15:IF C(I)<>4 THEN 560 ELSE N
EXT
1170 IF E$="1" THEN FOR I=16 TO 24:IF
C(I)<>4 THEN 560 ELSE NEXT ELSE FOR I=
22 TO 24:IF C(I)<>13 THEN 560 ELSE NEX
T:FOR I=16 TO 21:IF E$="2" THEN IF C(I
)<>4 THEN 560 ELSE NEXT ELSE IF C(I)<>
2 THEN 560 ELSE NEXT
1180 REM
1190 REM          DESPEDIDA
1200 REM
1210 SCREEN 3:IF E$="1" THEN A$="¡LIST
O!" ELSE IF E$="2" THEN A$="¡SABIO!" E
LSE A$="¡GENIO!"
1220 FOR I=2 TO 15:COLOR I:PRESET(20,8
5):PRINT#1,A$:FOR J=1 TO 500:NEXT:NEXT
:COLOR 4:SCREEN 1:LOCATE 4,11:PRINT "¿
OTRA PARTIDA? <S/N>"
1230 A$=INKEY$:IF A$="" OR INSTR("NnSs
",A$)=0 THEN 1230 ELSE IF INSTR("Ss",A
$) THEN 130 ELSE SCREEN 3:FOR I=2 TO 1
5:COLOR I:PRESET(20,85):PRINT#1,"¡ADIO
S!":FOR J=1 TO 500:NEXT:NEXT:NEW
```

Test de listado

10 - 0	130 - 36	250 - 0	370 -102	490 -238	610 - 0	730 - 0	850 - 0	970 -252	1090 - 11	1210 -203
20 - 0	140 -139	260 - 0	380 -117	500 -139	620 -124	740 -115	860 - 0	980 -100	1100 -230	1220 - 6
30 - 0	150 -191	270 - 0	390 -111	510 -192	630 - 0	750 - 37	870 - 0	990 - 0	1110 - 69	1230 -124
40 - 0	160 -142	280 -151	400 -220	520 - 13	640 - 0	760 - 2	880 - 37	1000 - 0	1120 - 77	
50 - 0	170 -182	290 -173	410 - 65	530 - 0	650 - 0	770 - 94	890 -204	1010 - 0	1130 - 0	
60 - 0	180 - 18	300 - 0	420 -247	540 - 0	660 -112	780 - 0	900 -235	1020 - 7	1140 - 0	
70 - 0	190 -210	310 -152	430 - 76	550 - 0	670 - 0	790 - 0	910 -167	1030 -142	1150 - 0	
80 - 0	200 - 83	320 - 97	440 - 38	560 - 85	680 - 0	800 - 0	920 - 0	1040 - 56	1160 - 55	
90 - 0	210 - 0	330 - 33	450 - 27	570 - 52	690 - 0	810 -246	930 - 0	1050 -169	1170 -226	
100 - 0	220 - 0	340 -179	460 -254	580 - 84	700 -154	820 -159	940 - 0	1060 - 0	1180 - 0	
110 - 67	230 - 0	350 -176	470 -190	590 - 0	710 - 0	830 - 71	950 - 82	1070 - 0	1190 - 0	TOTAL:
120 - 91	240 -189	360 - 44	480 - 73	600 - 0	720 - 0	840 -154	960 - 70	1080 - 0	1200 - 0	8466

MSX EXTRA

LA PRIMERA REVISTA DE MSX DE ESPAÑA
NUMERO ESPECIAL - P.V.P. 275 PTAS (incluido IVA)

Especial

Código Máquina

LA GIROMATICA (III)

FUNCIONES DE ANGULO DE GIRO

Comenzamos a ver en el artículo del pasado mes las figuras base, que como debéis saber no son otra cosa que el conjunto de puntos iniciales antes del primer giro. En el presente artículo, ilustrado con fotos del programa "giromatic" profundizaremos sobre las figuras base y además veremos las evoluciones de algunas figuras.

4.– Funciones del ángulo de giro

La línea 90 puede expresarse de forma distinta a como se ha hecho, ($Z=Z+H$), poniendo la expresión de Z en función de H y J sin introducir la propia variable en la expresión.

4.1– Giro proporcional

Una primera forma que cabe pensar es poner el giro de cada punto proporcional a su número de orden. Con ello la expresión de z sería:

$$Z=J.H$$

A este tipo de giro le llamaremos proporcional. Con la expresión indicada, el punto cero permanece siempre en la misma posición. Si queremos que gire todo el conjunto habrá que poner $Z=(J+1).H$.

4.2– Giro acelerado

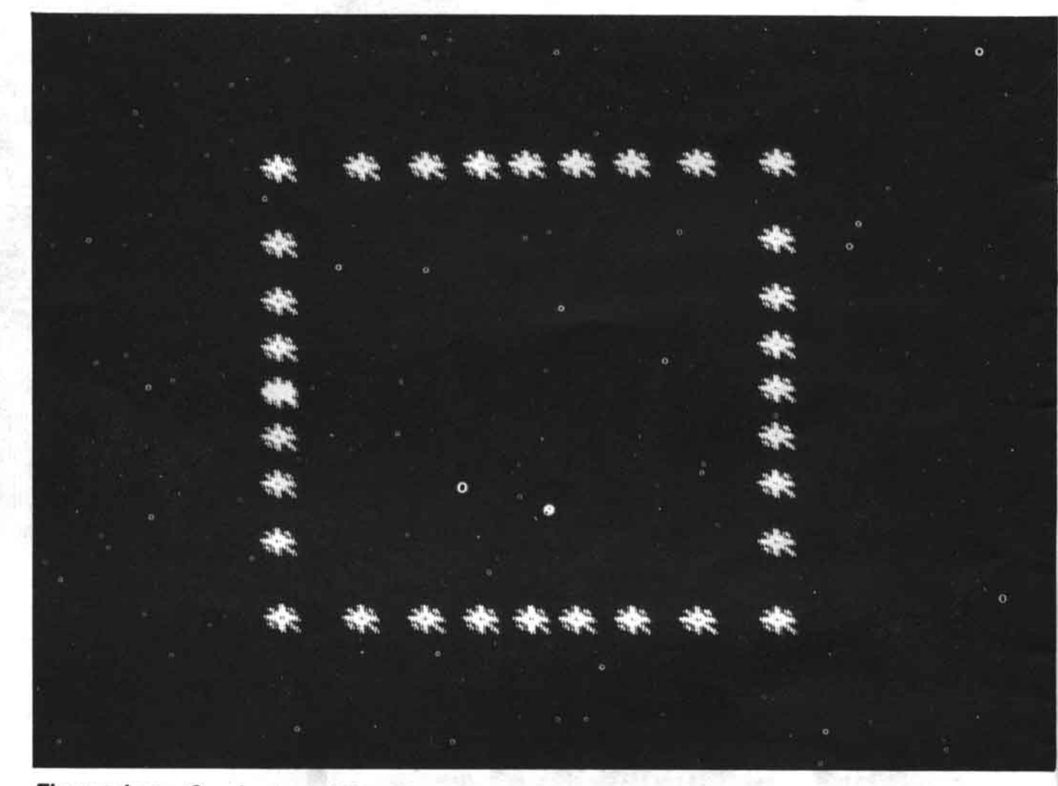
Si calculamos el giro provocado por la expresión $Z=Z+H$ podemos ver que el giro del punto J en la iteración M es:

$$Z=(J+1\ +(M-1).(N+1)).H$$

Para programar este giro, habrá que introducir en el programa la variable M que recuente las iteraciones o número de figura, para lo cual basta hacer $M=M+1$. Iniciando esta variable con el valor cero (que correspondería a la presentación de la figura base) se tendrá:

$$Z = (J+1+M.(N+1)).H$$

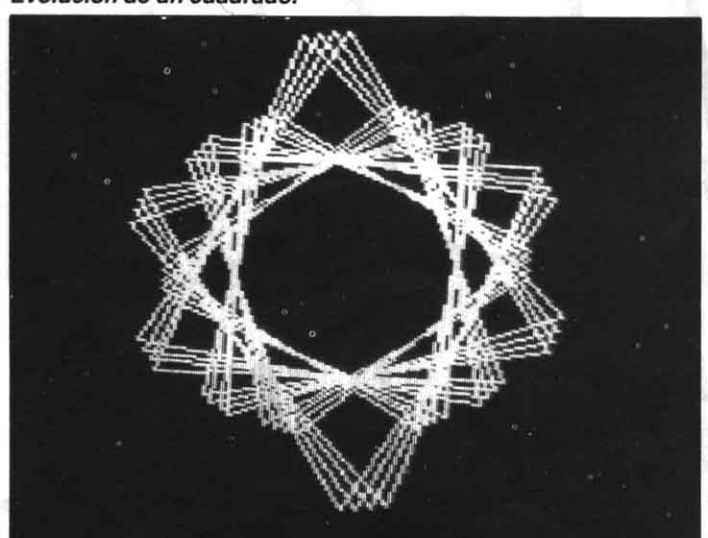
y si queremos que la figura no gire el



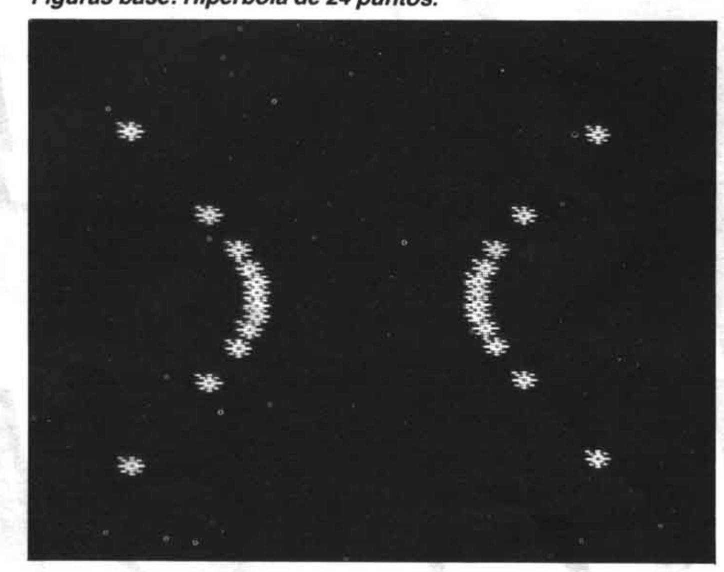
Figuras base. Cuadrado de 32 puntos.



Evolución de un cuadrado.



Figuras base: Hipérbola de 24 puntos.



punto O, la expresión sería:

$$Z = (J+M.N).H$$

El giro que produce esta fórmula es mucho mayor que el giro proporcional, por lo cual le llamaremos "giro acelerado", aumentando el ángulo de giro con el valor de M. Esta variable nos define el número de figura, que puede ayudar a identificar ésta en la gran mayoría de los casos, es decir cuando se haya mantenido constante el valor de H en una serie de figuras definidas por una figura base y un número de puntos N.

4.3- Giro simétrico

Una forma de hacer girar los puntos sería que lo hicieran alternadamente en sentidos opuestos y además proporcionalmente al número de figura. A este tipo de giro le llamamos "simétrico" y viene dado por

$$Z=H.(-1)^J.(M+1)$$

4.4- Giro trigonométrico

Una forma de giro que da buenos resultados, es el que llamaremos giro "trigonométrico", en el que el giro de cada punto se establece como función trigonométrica de su número de orden.

Para ello haremos:

$$S = \frac{2.Pi}{N}.Q \quad y\, Z=H.sen\, S.J.$$

o bien con las funciones coseno o tangente en lugar del seno.

Cuando J varía de O a N,S toma valores entre 0 y 2.Q.Pi, siendo Q una variable que nos define el número de vueltas o bucles establecidos para la función trigonométrica. Haciendo Q=O, la figura no gira con las funciones seno y tangente y gira un ángulo constante en cada vuelta con la función coseno, es decir tenemos siempre en la pantalla la figura base, quieta o girando.

5.- Figuras base en el programa "giromatic"

En el apartado 3 se trató de un caso elemental de figura base, constituida por los puntos de un segmento rectilíneo situados equidistantes en el eje polar. Con esta figura se ha podido experimentar la clave del programa, es decir, la forma en que se hacen girar los puntos.

Las figuras que se van consiguiendo a partir del segmento rectilíneo, son ya bastante originales, especialmente cuando se unen los puntos mediante líneas rectas, aunque por su sencillez no se incluyen en el programa "GIROMATIC". Sin embargo, es fácil su inclusión pues requiere pocas modificaciones.

A continuación se van a analizar figuras base más complejas, que dan vistosas figuras, y en algunas ocasiones muy curiosas, según los parámetros que se utilicen.

Elipse

Iniciamos la descripción con la elipse, que es la que dio motivo al invento de este programa, como ya se ha explicado en la introducción.

5.1- Elipse

Vamos a definir una elipse mediante la intersección de las proyecciones de los puntos de dos circunferencias concéntricas situados en el mismo radio, tal como se indica en la figura. Sea R el radio de la circunferencia mayor y E.R. el de la menor, siendo E menor que 1. Si el ángulo de radio con el eje polar ROX lo designamos por A, se tiene que las coordenadas del punto P son: x=R.cos A y=E.R. sen A cumpliéndose por tanto:

$$\frac{x^2}{R^2} + \frac{x^2}{E^2.R^2} = ;\ \text{ecuación de una}$$

Como que pretendemos expresar el giro de los puntos en coordenadas polares, habrá que determinar el módulo y argumento del punto P.

Si el módulo del punto P lo designamos por B(O,P) se tiene:

$$B(O,P)=\sqrt{x^2+y^2}=R.\sqrt{cos^2A+sen^2\ A.E^2}$$

El argumento POX que designaremos por B(1,P) será:

$$tg\ B(1, P) = y/x = E.\ tg\ A$$ y por tanto $$B(1, P) = arc\ tg\ (E.\ tg\ A)$$

Si queremos representar una elipse con N puntos, podemos hacerlo definiendo N ángulos iguales a A, con lo que el valor de A sería:

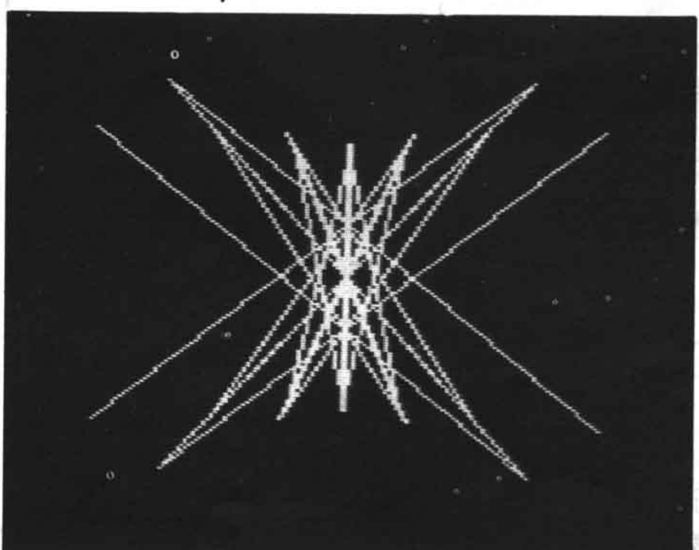
$$A = 2.\ Pi/N$$

Los N puntos quedan así situados en un ángulo de 360 grados. Pero pueden también repartirse en un ángulo mayor, dando K vueltas completas de circunferencia, en cuyo caso el ángulo sería:

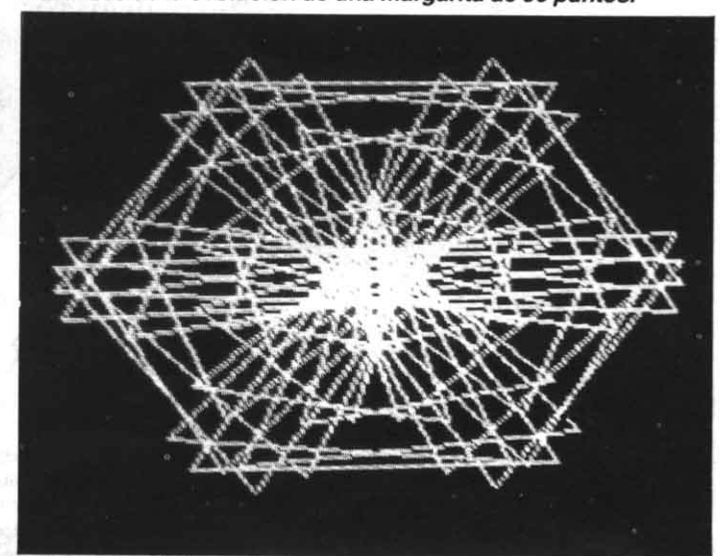
$$A= 2.\ K.\ Pi/N$$

El valor de K puede ser entero o fraccionario (decimal). En éste último caso, los N puntos quedan repartidos en un número no entero de vueltas.

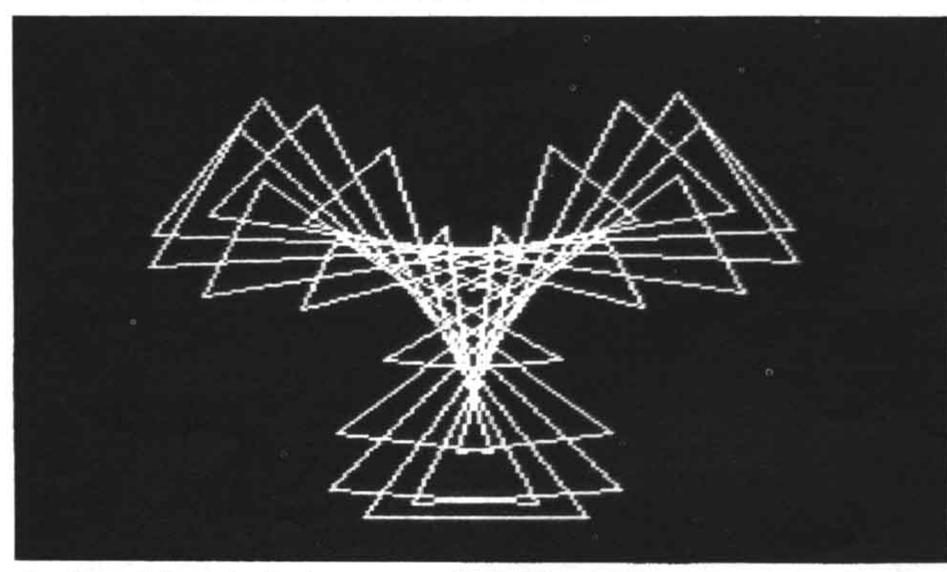
Evolución de una hipérbola.



Dos fases de la evolución de una margarita de 96 puntos.



LA GIROMATICA (III)



Dos fases de la evolución de una margarita de 96 puntos.

Los N puntos vendrán pues definidos por A_J =J.A., variando J de 1 a N. Sin embargo, como por comodidad definiremos para N puntos una matriz a partir del valor de N, en realidad tendremos N+1 puntos al variar en el programa J de O a N. Si el valor de K es entero, coincidirán en la representación gráfica, el primer y último punto.

Aunque los intervalos del ángulo A entre dos puntos consecutivos son iguales, no sucede lo mismo con el argumento del punto B(1, J). El sistema empleado para definir así la elipse, da lugar a que los puntos estén más concentrados en los extremos del eje mayor que del eje menor, es decir, hemos definido por puntos una curva con densidad variable de puntos a lo largo de su trazado. Este aspecto es muy importante en el prograama «GIROMATIC», ya que por su causa se producen figuras en las que se mezclan líneas rectas con curvas, por lo menos aparentemente en la pantalla.

En efecto, en las zonas de alta densidad de punto, aún a pesar de los giros que se van produciendo y que tienden a separar cada dos puntos consecutivos, si el ángulo de giro H es pequeño, los puntos sucesivos quedan muy próximos, de forma que al unirlos mediante las sentencias LINE, la apariencia en la pantalla es de líneas curvas. En cambio, en las zonas de baja densidad de puntos, éstos quedan suficientemente distanciados y por tanto las rectas aparecen como tales.

Por J.M. Climent Parcet

EN PANTALLA

HACKERS versus PIRATAS

Según varios informes realizados a nivel nacional en Japón, se está incrementando de forma alarmante el fenómeno HACKER. Existe en nuestro país un cierto desconocimiento de este fenómeno, que en numerosos casos se confunde con la piratería. En realidad se ha demostrado que en aquellos países en los que abundan los Hackers los piratas disminuyen.

Vamos a comentar someramente en qué consisten los hackers y los piratas.

Un pirata informático (tan abundantes por nuestros lares) es aquella persona que se dedica a desproteger y copiar ilegalmente programas comerciales de cualquier índole. En el mercado MSX destaca la enorme piratería en el campo de los juegos de ordenador, aunque también se produce con otros programas. Existen muchas clases de piratas: desde aquellos que copian los programas de su vecino/a conectando entre sí dos cassettes a los jóvenes genios de la informática que disfrutan rompiendo las complicadísimas protecciones de algunos juegos. Hay un abismo entre ambos.

Un hacker, por otro lado, es aquel que se dedica a introducirse en ordenadores ajenos. Esto es posible gracias a la interconexión de los grandes ordenadores mediante la línea telefónica. Mediante este sistema, con un pequeño ordenador, un MSX por ejemplo, el hacker puede introducirse en una red de grandes ordenadores tipo VAX o IBM. El daño producido por estos hackers puede llegar a ser enorme. Por suerte, en nuestro país los hackers son muy escasos, debido a la escasez de redes de ordenadores interconectados. Es este tipo de "incordios" informáticos el que está difundiéndose en Japón, dejando a un lado a los piratas. ¿Quién va a copiar un pequeño programa para MSX pudiendo utilizar grandes programas en ordenadores mucho mayores?

IBM ANUNCIA EL PC-ET

IBM anunció la aparición de su nuevo ordenador IBM PC-ET (Tecnología extendida), basado sobre un chip Intel 80386.

La importancia de esta aparición para los usuarios de MSX es relativamente grande, ya que, en su versión base, el ordenador viene dotado de un lector de diskettes de 3.5 pulgadas y de 720 Kb de capacidad. Este formato es el habitualmente utilizado por los MSX debido a sus ventajas respecto a los discos de 5.25 pulgadas.

Esta decisión de IBM marca definitivamente la pauta para la estandarización del nuevo (no tanto para los usuarios de MSX) formato de discos flexibles. El PC-ET incorpora en su versión básica un lector de diskettes de 3.5 pulgadas, como ya hemos comentado; y un disco duro de 30 Mbytes. Su resolución gráfica ascenderá a 1024×1024 píxels (puntos) generada por un chip 34010 de Texas Instruments. El teclado, por último será el teclado extendido del AT3, que incorpora, entre otros muchos detalles teclas separadas de control de cursor.

El sistema operativo que utilizará este aparato será un MS-DOS 5.00, o ADOS (recuérdese que los MSX utilizan el MSX-DOS, que desgraciadamente no alcanza el nivel de la versión 1.25 de MS-DOS).

Digamos, además que la memoria RAM del ordenador oscilará entre 1 Mbyte y 1 Gigabyte.

Repetimos por tanto; IBM adopta el sistema de unidades de disco hace tiempo utilizado por los MSX, con lo que los MSX serán capaces de cargar los discos generados con un IBM. Por fin se hará realidad aquello de repasar con el ordenador doméstico el trabajo realizado con el ordenador profesional, sin ninguna incompatibilidad.

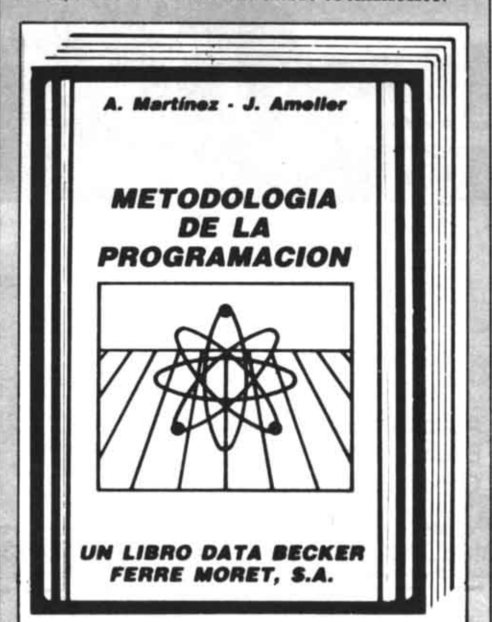
METODOLOGIA DE LA PROGRAMACION

Un libro muy útil para iniciarse

Ferré Moret, S.A. siguiendo la línea que le caracteriza ha editado un libro especialmente útil para aquellos que deseen iniciarse en las tareas de la programación.

Esta vez se trata del libro titulado "Metodología de la programación", recomendado especialmente a las escuelas de enseñanza de informática además de para neófitos. Este libro ha sido realizado por A. Martínez y J. Ameller profesores del Instituto politécnico del ejército español y de la Universidad nacional de educación a distancia.

Se trata de una introducción a la lógica del ordenador, que sirve como introducción a la programación en cualquier lenguaje. No se requieren conocimientos previos de informática, y además contempla diversos métodos de programación, desde los clásicos a los más modernos. En resumen, un libro recomendable totalmente.



SPACE SHUTTLE



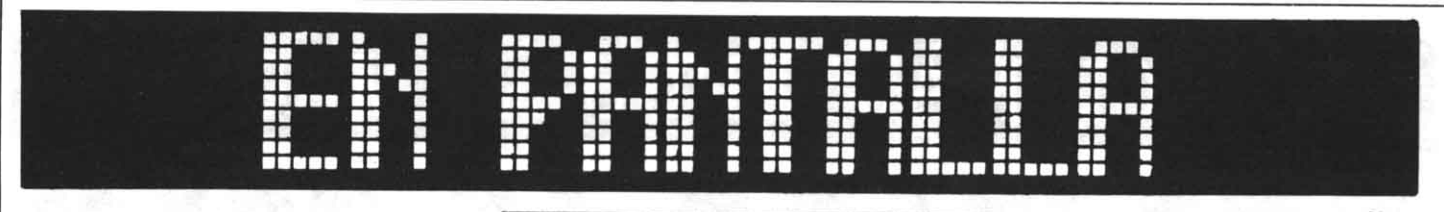
a journey into space

Este nuevo juego de ACTIVISION simula una operación espacial a bordo de una lanzadera espacial (Space Shuttle). Tu misión se divide en varias etapas: salir de la atmósfera terrestre, situarte en la órbita preestablecida, lanzar un satélite altamente secreto y tomar contacto con otros satélites.

Debes también realizar correctamente todas las maniobras de reentrada en la atmósfera para no quedar carbonizado en el intento. Por último has de aterrizar la lanzadera en la base de Edwars de la Fuerza Aérea.

El juego tiene la apariencia de un simulador de vuelo; pero resulta mucho más complejo. Las escenas son bastante más variadas debido a las diferentes fases del juego. Los gráficos y controles de la lanzadera están muy bien resueltos, ya que según se nos informa en el manual del programa, han sido realizados en colaboración con personal de la NASA, para conseguir el máximo realismo en el juego.

Los controles son numerosos y requieren un buen período de aprendizaje; pero los amantes de los juegos de simulación lo encontrarán interesante y extremadamente atractivo. Hagamos notar que, los



controles, siendo en apariencia similares a los de otros simuladores de vuelo no son, en realidad nada parecidos, debido a las condiciones de ingravidez a las que se ve sometida la nave.

Existen tres niveles en el juego. El primero es el de autovuelo, en el que pilotas una nave controlada desde tierra. De este modo puedes habituarte a los controles de tu nave sin peligro. Es un buen método para conseguir el dominio necesario de los controles, ya que en este nivel se pasa por todas las fases del juego.

En el segundo, el control de tierra solo maneja ciertos aspectos secundarios de la nave, de modo que la responsabilidad está en tus manos. Es el nivel idóneo para ampliar tu dominio de los controles y para aprender a solventar las posibles contrariedades que aparezcan a lo largo del vuelo.

Por último, en el tercer nivel, estás sólo ante los controles de tu nave. Tu única ayuda será el computador de a bordo que te avisará cuando las cosas empiecen a ir mal...

INFORMATICA EN ESPAÑA

Los pequeños se engrandecen

Según un informe realizado por la Dirección General de Electrónica e Informática del Ministerio de Industria el número de ordenadores instalados en nuestro país, entre 1983 y 1985, ascendió un 36,3 por 100, mientras que su valor sólo se incrementaba en un 7 por 100.

De este hecho se deduce, en primer lugar, un importante auge de la informática de bajo costo, es decir, los ordenadores personales y domésticos (sector este último en el que se inscriben nuestros MSX). Esta alza continuada del número de ordenadores personales y domésticos hace prever nuevas y más interesantes apariciones en este campo. Las mayores empresas de informática entran, cada vez con mayor fuerza en el campo de los ordenadores personales y domésticos. Claro ejemplo de esto son dos grandes empresas del mundo informático. Philips, empresa ampliamente conocida gracias a sus aplicaciones específicas controladas por grandes ordenadores se encuentra dentro del mercado MSX con numerosos modelos. Por otra parte, para dar un ejemplo algo más alejado de nuestro sistema, IBM, indiscutible líder en el sector informático, desarrolló un ordenador doméstico, el PC junior (PCjr), compatible con sus ordenadores personales de gama PC/XT/AT; pero destinado al mismo mercado que aquí ocupan los MSX.

Es bien patente, por tanto, la importancia que están desarrollando los pequeños ordenadores, y el interés de las grandes empresas por controlar este codiciado mercado.



ENTREGA DE PREMIOS 2.º GRAN CONCURSO DE PROGRAMAS

Recientemente tuvo lugar en nuestra redacción la entrega de los premios correspondientes al segundo gran concurso de programas convocado por nuestra revista.

Como sabéis, este premio correspondió a Eduard Playá González por su programa MONTGOLFIER. La entrega del mismo –una unidad de disco PHILIPS VY 0010– fue efectuada por nuestra directora ejecutiva Birgitta Sandberg.

Aprovechamos la entrega del premio para conversar con Eduard acerca de su programa y de la técnica que había empleado para su confección.

A la hora de confeccionar el juego, ¿tuviste problemas con los SPRITES?

–Más que problemas, trabajo. Para que el juego no tuviera problemas con la inexorable regla del quinto sprite, tuve que controlar las rutinas que mueven a los pájaros de modo que no me coincidieran más de cinco en una línea.

¿Qué técnica empleaste para optimizar la velocidad de ejecución del juego?

–Para mejorar la velocidad, tuve que acortar hasta el máximo posible el bucle principal, además de evitar que se produjeran al mismo tiempo las llamadas a las subrutinas. Si observáis el desarrollo del programa veréis que las llamadas a subrutinas se producen de una manera intermitente. A efectos de optimización, utilicé todos los trucos que publicáis en "Trucos de programador" referidos a ese punto. Me han servido de mucho. (Nos va a hacer enrojecer este chico).

¿Qué fue lo que más trabajo te costó a la hora de desarrollar el programa?

–Sobre todo el acortar el listado al máximo, puesto que las primeras versiones del programa tenían una longitud kilométrica, y la depuración del programa me llevó no poco tiempo. Además, como pretendí cuidar los detalles al máximo (instrucciones, títulos etc.) esta tarea de depuración me entretuvo bastante.

Y, hablando de música, ¿cómo ideaste la melodía?

–De oído, con los rudimentos de música que poseo. Cogí la flauta y empecé a probar hasta que conseguí una serie de notas que me gustaron, y realicé su transcripción. De todos modos tuve bastantes problemas a la hora de compatibilizar la música generada mediante SOUND y PLAY.

Y ahora, la pregunta inevitable. ¿Qué te parece el premio?

–Indiscutiblemente, estoy muy contento. Llevaba un temporada rondando una unidad de disco, además, tengo entendido que las unidades de PHILIPS son de lo mejorcito que hay para MSX. Efectivamente, Eduard no se equivoca. Las unidades PHILIPS VY 0010 unen a su solidez, excelente diseño y prestaciones, la garantía de su propia marca, sobradamente reputada no sólo por los usuarios de MSX, sino por cualquier sector del público general.

Tomen nota los futuros concursantes: concreción, claridad y velocidad de ejecución combinados con una idea original pueden darles el premio del próximo concurso.

DEL HARD AL SOFT 22

SISTEMAS DE COMUNICACIONES

En esta sección ya tradicional de nuestras páginas abordamos un tema por demás interesante. Los sistemas de comunicaciones y sus elementos básicos de interacción.

ELEMENTOS PRINCIPALES QUE INTERVIENEN EN UN SISTEMA DE COMUNICACIONES

Un sistema de comunicaciones se puede definir como un medio técnico que nos permite el intercambio de información y el tratamiento de la misma. En términos reales, nos encontramos en la actualidad con distintos ejemplos en el empleo de los sistemas de comunicaciones, aplicados a las más variadas manifestaciones. Los bancos, por ejemplo, utilizan continuamente estos medios; ingresar o extraer dinero da pie a que se inicie un proceso de consulta y respuesta, en el que se ve inmerso por completo el sistema de comunicaciones.

Históricamente los sistemas de comunicaciones han variado la técnica transmisora de la información. Al principio se hacía aleatoriamente, después, mientras el hombre se civilizaba, fueron cambiando las técnicas: el humo de los indios americanos, las señales con espejos, el telégrafo, hasta llegar al actual conglomerado compuesto por los ordenadores.

Según sea el tipo de utilización del sistema de comunicaciones, podemos agruparlos en los siguientes apartados:

–CONSULTA.– La consulta es un servicio ordinario que prestan los servicios de comunicaciones. A través de la consulta, la persona que la haya hecho, puede acceder, a través de una terminal, a los archivos o la base de datos. Esta devuelve la información pedida en un proceso muy rápido de búsqueda.

–ACTUALIZACION DE REGISTROS: La actualización de registros permite actualizar el registro de la base de datos, aunque se produzca desde una terminal a distancia al ordenador



central o principal, pudiendo alterar de esta manera la información registrada. Por ejemplo, será necesario alterar los registros, en las operaciones de los bancos cada vez que el gobierno cambie el tipo de imposición fiscal.

–ENTRADA DE DATOS.– La operación de entrada de datos solo permite la entrada de información al ordenador central, sin que se tenga que dar una respuesta o haya que mantener abierta la base de datos para otras operaciones. En el caso de la entrada de datos, la información llega directaamente al ordenador central y éste la trata automáticamente.

–TIEMPO COMPARTIDO.– A través del tiempo compartido se permite el tratamiento de información a varios usuarios a la vez. Esta es una aplicación, llamada también proceso interactivo, que además de recibir la información de varias terminales, puede dar respuesta a todas ellas de manera que al usuario le parece estar haciendo uso exclusivo del ordenador.

–ENTRADA DE TRABAJOS A DISTANCIA.– Esta aplicación dota al usuario de la posibilidad de enviar información desde una terminal a distancia y recibir respuesta. Se subdivide en dos tipos de procesos (RJE) (Entrada remota de trabajos por lotes) y CRJE (Entrada remota de trabajos por lotes de tipo conversacional). En la primera, el usuario envía sus datos y los resultados se le devuelven a su terminal a distancia. En la segunda, se produce una conversación entre la terminal y el ordenador central, permitiendo correcciones sucesivas del trabajo enviado.

–CONMUTACION DE MENSAJES.– Permite enviar mensajes desde un punto de la red a otros puntos de la red aunque esten diseminados en zonas muy amplias. Es un proceso muy utilizado para dar los resultados de unas votaciones generales o de intercambio de información entre grandes empresas. Normalmente, para este tipo de aplicación se utilizan redes de comunicación muy extendidas.

Vista la utilización posible de los sistemas de comunicaciones, podemos deducir las siguientes características:

–ENTRADA Y SALIDA A DISTANCIA. A través de los servicios de comunicación los recursos y respuestas del ordenador central circulan comunicándose con las terminales a distancia.

–ENTRADA AL AZAR. Cualquier terminal a distancia puede acceder al ordenador central a cualquier hora.

–PROCESO INMEDIATO DE TRANSACCIONES. Permite el proceso de datos inmediato, no teniendo que esperar a que los «lotes» de información se pro-

cesen todos juntos. De esta manera cada transacción se procesa individualmente en el acto.

–TIEMPO DE RESPUESTA RAPIDO. Los sistemas de comunicaciones están diseñados para ofrecer tiempos de respuestas lo más rápidos posible.

–MULTIPLES USUARIOS SIMULTANEOS. Como hemos dicho anteriormente, este sistema permite que se acceda al ordenador central a través de varias terminales trabajando al mismo tiempo.

–«ON-LINE». A través de este término, se define la operación de entrada de datos desde la terminal a distancia al ordenador central.

–TIEMPO REAL. Este término se utiliza para calificar a las operaciones que tienen una respuesta suficientemente rápida, y por lo tanto permite la corrección de errores inmediatamente sin que se tenga que esperar a que se cumplan otras operaciones que puedan producir, en algunos casos, dificultades de comunicación.

ELEMENTOS BASICOS DE LOS SISTEMAS DE COMUNICACIONES

Los ejemplos básicos de los sistemas de comunicaciones son: El ordenador central o principal, los medios y servicios de comunicaciones y las estaciones de terminal a distancia.

Las terminales a distancia suponen el acceso al ordenador central y actúan, también como medio para poder extraer información del mismo. La forma de la terminal, varía según sea su aplicación. Normalmente el equipo de la terminal consta de un teclado con un registro impreso y salida impresa.

La función de los medios y servicios de comunicaciones es la de hacer llegar el mensaje hasta el ordenador central, y en el caso de que se produzca respuesta, devolverla a la terminal.

Aparte de proporcionar el programa y la base de datos, el ordenador central, ha de estar preparado para reaccionar ante cualquier demanda solicitada por una de las terminales. Por lo tanto no es posible que el ordenador central funcione solo con procesos de entrada/salida por lotes rutinarios. Normalmente se ha de utilizar un NCP (Programa de control de redes).

El NCP es un sistema operativo cuya función es la de control y tratamiento de la red de comunicaciones.

El interfaz o «interface» es el punto donde se encuentran los procesos de datos con el sistema de comunicaciones. Los diseñadores son los encargados de engendrar los dispositivos capaces de realizar las operaciones necesarias en el interfaz. Estos dispositivos reciben el nombre de unidades de control de comunicaciones.

Si extendemos los tres elementos principales de los sistemas de comunicaciones a estos dos últimos podemos constatar que los elementos principales de un sistema de comunicaciones son cinco:

–Estación terminal a distancia.

–Medios y servicios de comunicaciones.

–Ordenador central.

–Programación de sistemas de comunicaciones.

–Unidades de control de comunicaciones.

AREAS DE RESPONSABILIDAD

Normalmente podemos considerar responsables en un proceso de datos del éxito de la operación, al usuario y al fabricante. El usuario es responsable de la buena utilización de la terminal y el fabricante es el responsable del buen funcionamiento del sistema. Sin embargo, en las aplicaciones donde se han de transmitir datos a grandes distancias es la compañía telefónica la responsable del envío y la recepción de datos.

La compañía telefónica aunque no interviene en la programación de los datos, ha de asegurar la entrega de los datos entre lugares separados por grandes distancias.

Como es de suponer, la comunicación de datos a través de la compañía telefónica, no se hace a través de mensajes hablados, sino mediante señales digitales eléctricas.

Juan Carlos González.

MEDIOS DE COMUNICACION

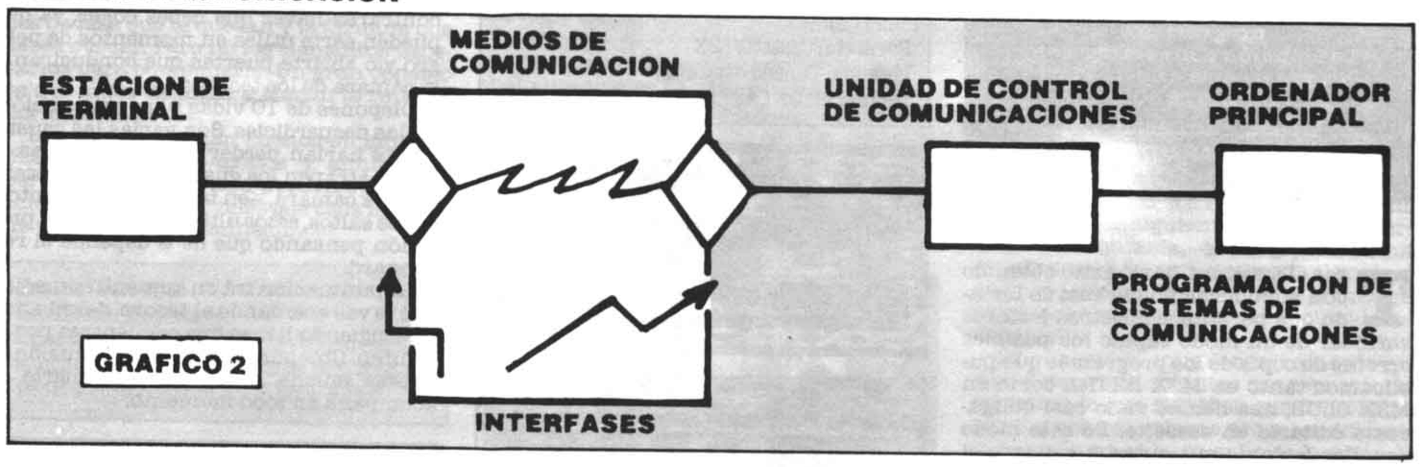


GRAFICO 2

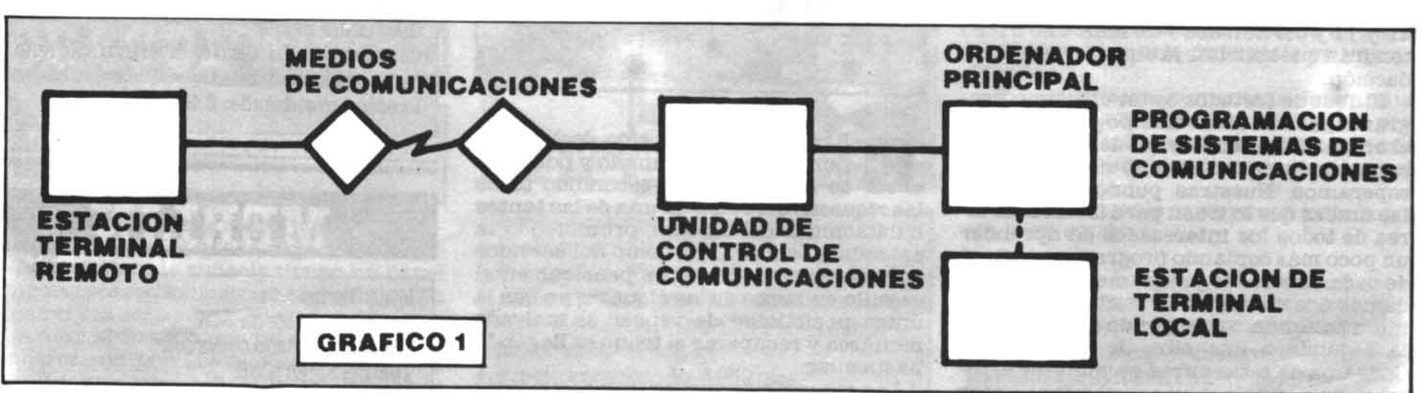


GRAFICO 1

BIT-BIT Software Juegos

POR CLAUDIA TELLO HELBLING

TEST DE LISTADOS

Manhattan Transfer, S.A.
Formato: Cassette 32K
Mandos: Teclado
Carga: Merge "CAS:"



Tras el extraordinario éxito obtenido con la publicación del Test de Listado que permite a nuestros lectores verificar de un modo rápido los posibles errores de copia de los programas que publicamos tanto en MSX EXTRA como en MSX CLUB, nos hemos visto casi obligados a editarlo en cassette. De este modo aquellos lectores que quieran evitarse el tener que copiarlo de nuestros números 10 y 11 y del número 7 de MSX CLUB sólo tendrán que solicitar la cinta a nuestra redacción.

El Test de Listados es un utilísimo programa para ahorrar tiempo y evitar tirar el ordenador y acordarnos de muchas personas no queridas cuando no sale lo que esperamos. Nuestras publicaciones son las únicas que lo traen para facilitar la tarea de todos los interesados en aprender un poco más copiando programas. Al final de cada uno de los programas publicados damos dos columnas de números. La primera columna corresponde a las líneas y la segunda a una cifra de verificación. Cada una de estas cifras se suman y el total tiene que coincidir con el que el lector obtenga tras emplear el Test de Listado. Si la suma coincide el programa tiene que funcionar correctamente. Si coincide y no funciona, puede suceder que haya algunas diferencias compensadas en algunas líneas que son fáciles de localizar. Del mismo modo si la cifra total no es igual a la que publicamos el lector tiene que buscar en qué línea se encuentra el error. Una vez localizada vuelve al programa de la revista y verifica que la copia sea correcta. Su empleo es muy fácil.

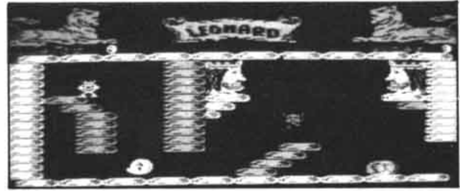
Es muy importante que no renumere nuestros programas hasta que los haya verificado ni haga ningún tipo de modificación, pues de hacerlo así el Test no le servirá de nada.

Conclusión: Programa muy útil para los lectores de MSX EXTRA y MSX CLUB. Sólo se vende en n/redacción o por correo.
Precio aproximado: 500 ptas.

KING LEONARD

Mind Games
Formato: Cassette 32K
Mandos: Teclado o joystick
Teclee: Bload "CAS:",R





El rey Leonard ha sido destronado por su propio hermano y por si esto fuera poco ha escondido todas las riquezas del reino en una de las tantas habitaciones del castillo, prohibiendo la entrada a Leonard. Tú, como fiel servidor del rey destronado, debes penetrar en el castillo en busca de las riquezas ya que la única posibilidad de vencer al malvado monarca y recuperar el trono es llegando hasta ellas.

Las riquezas se encuentran escondidas



en "la cámara de los tesoros" pero una vez dentro del castillo deberás pasar por todas las habitaciones. Has de tener mucha precaución con los guardias que vigilan celosamente la cámara pero también hazlo con los que vigilan cada aposento pues su maldad iguala a los anteriores. Si te siguen procura despistarlos o estarás perdido.

En diferentes rincones del castillo encontrarás llaves que debes coger, ya que pueden serte útiles en momentos de peligro y/o abrirte puertas que conducirán a la cámara de los tesoros.

Dispones de 10 vidas para esta misión, no las desperdicies. Son varias las causas que te harían perder una o más vidas: —que te atrapen los guardias de los aposentos o la cámara, —en tus desplazamientos, en los saltos, estos últimos hazlos con precisión pensando que de ti depende el rey Leonard.

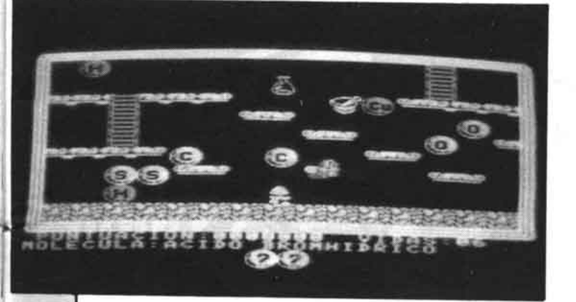
La puntuación irá en aumento a medida que te vas acercando al tesoro, derribando y recogiendo llaves que encuentres por el camino. Obtendras éxito en esta misión si posees mucha habilidad y la suerte te acompaña en todo momento.

Sonido: Convencional.
Grafismo: Bueno.
Conclusión: Un juego de laberinto sólo apto para jugadores hábiles.
Precio aproximado: 2.495 ptas. + IVA

MERLIN

Mind Games
Formato: Cassette 32K
Mandos: Teclado o joystick
Teclee: Bload "CAS:",R

Seguro que todos conocéis al gran mago Merlín, famoso por sus hechizos para ayudar a los demás. Ahora, nuestro amigo se encuentra en un verdadero problema y necesita de tu ayuda. Resulta que está prisionero y en la parte más deshabitada del castillo donde nadie puede oírle. Sin embargo, éste no es su principal inconveniente. Todo comenzó cuando Merlín estaba preparando un hechizo y por alguna equivocación en los ingredientes comenzaron a aparecer extraños seres que desde entonces no cesan de acosar



al pobre mago, obstaculizando sus experimentos.

Para acabar con estos seres malignos Merlín ha de conseguir las fórmulas. Pero otra trastada que le han hecho los malditos es esparcir todas las fórmulas y mezclarlas por todos los rincones del castillo. Entre estas fórmulas se encuentra la solución para derrotar a los genios del mal. Pero no sólo han revuelto papeles y escondido el libro de Merlín sino también los ingredientes que éste puede usar. El juego consiste pues en encontrar la fórmula que destruirá a los seres y también los ingredientes para realizarla.

En la pantalla tienes un indicador de la puntuación que vas obteniendo a medida que avanzas en el juego, de las oportunidades o vidas que te quedan y la fórmula que debes buscar. Pero no los elementos necesarios para componerla. Un gran consejo es que antes de jugar a Merlín te repases las formulaciones.

Los que no son tan buenos en química pueden elegir los óxidos y anhídridos, pero los más atrevidos pueden optar por ácidos y sales. No olvides que cada fase tiene un tiempo determinado.

Sonido: Correcto.
Grafismo: Muy bueno.
Conclusión: Además de entretenido resulta sumamente útil para saber más de química.
Precio aproximado: 2.495 ptas. + IVA

DRAGONWORLD

Idealogic/Philips
Formato: Diskette p/MSX 2
Mandos: Teclado

Este es un magnífico juego de aventuras en el que la capacidad de decisión y la audacia tienen un papel importante en la consecución del éxito. Tu misión y la de tu amigo Hawkwind es la de rescatar al Dragón que ha sido raptado y encarcelado por las fuerzas del mal. Tu nombre es Amsel y eres un científico muy especial, pues no sólo te vales de las cosas racionales, sino también de las fantásticas para conseguir tu propósito. Esto significa que si no te sirve la espada, un barco u otro objeto puedes echar mano a determinadas palabras mágicas para continuar tu camino hacia las profundas tierras del Sur de Simbala, donde está prisionero el Ultimo Dragón.

Para jugar te recomiendo que una vez cargado el disco prepares otro a fin de grabar en él el camino que hayas recorrido y no tengas que volver desde el principio. Para crear un disco de juegos tiene que emplear la orden PREPARAR, cuando el peligro de destrucción es inminente, y pulsa RETURN. De modo que si te destruyen puedes restablecerte en el mismo punto. Para hacerlo tienes que usar la orden RECUPERAR.

Como ya es habitual en este tipo de juego no debes complicarte la vida dando órdenes confusas. En principio trata de colocar el verbo primero seguido del objeto, las características de éste. También puedes determinar en una misma orden varias órdenes, pero éstas deben ser coherentes y secuenciales. Por ejemplo no puedes zarpar sin antes subir al barco, ni leer un libro sin antes cogerlo y abrirlo. Los adjetivos o sustantivos que emplees y que no sean aceptados te aparecerán en pantalla como vocablos que no están en el idioma de Simbala. El mismo criterios se sigue cuando te encuentran con alguien y hablas. Quiero decir que das la orden de hablar, preguntar y después especificas con quién y sobre qué o quién.

Uno de los recursos mágicos que tienes para salvarte de algún peligro es invocar la Perla del Dragón. Ella te sacará de apuros y podrás recomenzar la búsqueda desde un punto más seguro. Pero todo tiene su límite y no podrás invocarla más de cinco veces. También puedes pedir los inestimables consejos de tu amigo Hawkwind para lo cual tienes que localizarlo allí donde esté.

Este juego desarrollado por Idealogic, bajo licencia de Telarium, ha contado con un equipo totalmente español, integrado por Norbert Martínez, Ricard Gavaldà, Eloi Serra, Agustín Pérez, Eloi Ortega, Joan Carmona y Josep M.ª Losantos. También han participado en las ilustraciones Romá Gutiérrez, Oscar Eroles, Susanna García y Cristina Carré.

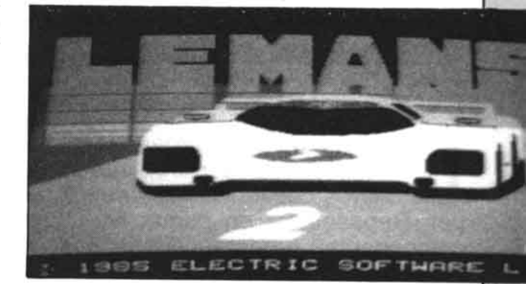


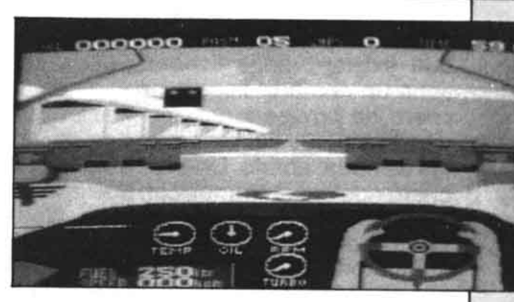


Grafismo: Excelente.
Sonido: Correcto.
Conclusión: Un magnífico juego de aventuras desarrollado para MSX-2 aprovechando la potencia de estos aparatos y propiciando la imaginación del jugador.
Precio aproximado: 5.950 ptas.

LE MANS 2

Serma
Formato: Tarjeta Softcard
Mandos: Teclado o joystick





Para los aficionados a fórmula 1 os presentamos el juego ideal; una versión de las famosísimas "24 horas de Le Mans", pero al adaptarlas a vídeo juego se ha tenido que variar un poco. Esta diferencia consiste en que las 24 horas pasarán a ser 24 minutos, lo que es una gran ventaja para vosotros, ¿no creéis?

Para dar por terminada la carrera se han de hacer las 10 vueltas que componen "Los 24 minutos de Le Mans".

Si te has decidido a participar, tendrás a tu disposición un hermoso coche deportivo. Tu puesto es el de conductor así tendrás una perspectiva más real y emocionante, con un grafismo tridimensional. La luz verde indicará la salida; el coche puede llegar a alcanzar los 360 km/h y sus funciones son las siguientes: –girar a la derecha, –a la izquierda, –acelerar, –disminuir la velocidad y frenar apretando el botón de disparo; está equipado con 5 marchas que cambiarán automáticamente cuando sea necesario.

La carrera es muy larga, recuerda que su límite son 24 minutos que representan 24 horas lo que significa que deberás pasar la noche al volante procurando que el cansancio no te haga salir de la pista. La carrera la forman 20 participantes, si no quieres que te penalicen procura no llegar el último, ni salirte de la pista en una derrapada.

Consejos para aumentaar la puntuación: –mantener el coche en la pista procurando el mínimo de accidentes, –adelantar a los demás coches y sacando vueltas de ventaja.

Para finalizar la carrera con éxito busca un buen copiloto.

Sonido: Bueno.
Grafismo: Visión tridimensional de la pista.
Conclusión: Juego emocionante.
Precio aproximado: 5.200 ptas.

LOS FICHEROS EN MEMORIA

Una de las razones que han impulsado el desarrollo de la informática en los últimos años es la facilidad con que los ordenadores tratan grandes volúmenes de datos. En este artículo veremos una de las herramientas fundamentales en el tratamiento de datos: los ficheros.

Los ficheros en los ordenadores tienen una gran similitud con los ficheros convencionales sobre papel. Repasemos, en primer lugar, los componentes fundamentales de un fichero convencional.

En primer lugar contamos con el fichero, el conjunto de fichas. En caso de que existan varios ficheros, cada uno tiene un nombre: fichero de clientes, fichero de proveedores, etc. Para acceder a los datos del fichero debemos, antes de nada, abrirlo. Una vez abierto el fichero nos encontramos con que la información está dividida en fichas. Cada una de estas fichas tiene un código o un nombre según el cual están ordenadas en el fichero. Una vez tenemos la ficha ante nosotros observamos que está dividida en varios campos: nombre, dirección, etc. Para acabar, una vez retornada la ficha a su sitio, cerramos el fichero.

Hemos expuesto este proceso de acceso a la información de una ficha en un fichero convencional, ya que el sistema utilizado para obtener la información en un fichero de ordenador es prácticamente el mismo.

Distinguiremos antes de entrar en materia, tres tipos de ficheros, cuyo modo de acceso será bien diferente. El primero de ellos es el fichero en memoria: este fichero está incluido totalmente en la memoria del ordenador, y se pierde al apagar el mismo. El segundo tipo de fichero es el secuencial. Secuenciales son los ficheros en cassette, ya que la información se almacena un dato tras otro, aunque también podemos generar ficheros secuenciales sobre el disco. En estos ficheros sólo podemos acceder a un dato si hemos pasado por todos los anteriores. Por último están los ficheros de acceso aleatorio, o ficheros «random». Estos ficheros, sólo direccionables desde el disco, permiten obtener un dato de forma directa.

Trataremos con calma de todos ellos, y daremos ejemplos de cómo se pueden crear y utilizar. Empezaremos con los más sencillos: los ficheros en memoria.

FICHEROS EN MEMORIA

No existe una estructura especial en BASIC para generar ficheros en memoria (sí por ejemplo en PASCAL), por lo que, para almacenar los datos utilizaremos conjuntos de cadenas de caracteres. Repasemos en primer lugar las estructuras BASIC necesarias para el desarrollo de los ficheros en memoria.



DIM: El comando DIM genera un conjunto ordenado (una única dimensión) o tablas (más de una dimensión) según se lo especifiquemos. Por ejemplo:

DIM nombre$ (100)

genera un conjunto, llamado nombre$, que contiene 100 cadenas de caracteres (por llevar dólar). En cambio,

DIM edad (145)

genera un conjunto, llamado edad, que contiene 145 números.

Para colocar un 7 en el elemento 43 del conjunto edad haríamos:

edad (43) = 7

A algunos de los lectores esto les parecerá trivial y extremadamente simple, a otros, por el contrario, les sonará a chino. Intentaremos contentar a todos.

Como hemos dicho al principio, en una ficha diferenciamos la información en varios campos: nombre, edad, etc. En los programas que utilicen ficheros en memoria, asignaremos cada campo a un conjunto. Tomaremos como ejemplo, a partir de ahora, una ficha que sólo incluya nombre, edad y número de teléfono, para simplificar, aunque el proceso es el mismo si se desean utilizar más campos.

Nuestro programa debería comenzar con:

DIM nom$(100), edad(100), tlf$(100)

Hemos abreviado nombre por nom, así que el nombre se almacenará en nom$, y asímismo teléfono por tlf. Hemos asignado al teléfono un conjunto de cadenas de caracteres. Hemos hecho esto para facilitar la entrada de los teléfonos con guiones o puntos de separación, que no se podrían utilizar si hubiésemos declarado tlf sin el signo de dólar.

El número 100 representa el máximo número de fichas que podrá almacenar nuestro programa (100 está bien por el momento). El acceso a la ficha se consigue modificando el nombre, la edad y el teléfono de un mismo índice al mismo tiempo. Quedará más claro con un ejemplo:

```
nom$ (1)="Pedro Pókez Remírez"
edad (1)=37
tlf$ (1)="(23)-34-43-12"
```

con esto sabemos que el nombre de la ficha uno es Pedro..., la edad de la ficha uno es 37 y el teléfono de la ficha uno es (23)-34-43-12. Con esto he-

mos realizado un alta en nuestro incipiente fichero.

Pero lo más interesante de un fichero es poder acceder a sus datos. Nada más fácil. Si queremos ver el teléfono de la ficha uno no tenemos que hacer más que

```
PRINT tlf$ (1)
(23)-34-43-12
```



Para eliminar fichas podemos utilizar varios métodos. El primero, y más evidente, es desplazar todas las fichas que hay detrás un puestro atrás. Por ejemplo, si quisiéramos borrar la ficha 33 haríamos

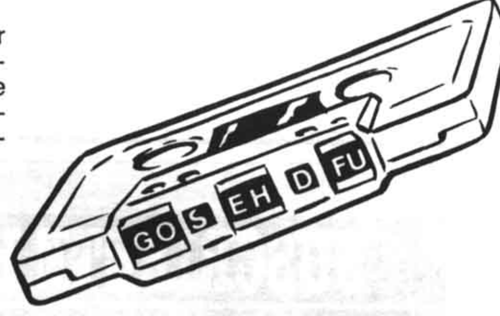
```
FOR x =33+1 to 100
nom$ (x-1)=nom$ (x)
edad (x-1)=edad (x)
tlf$(x1)=tlf$(x)
NEXT x
```

Este sistema tiene un grave inconveniente si el fichero es muy grande: la lentitud. Si queremos borrar la ficha 1 en un fichero de 5000 fichas tendremos que desplazar 4999 fichas. Existen otros sistemas para evitar esto, siendo el principal el de la indexación, de la que hablaremos más adelante. Otro ardid muy utilizado es introducir un carácter especial para indicar que la ficha está vacía. Por ejemplo, si el primer carácter del nombre es un "$", podemos suponer que la ficha está vacía. Quede claro que esto lo hacemos por conveniencia; no hay ninguna obligación de utilizar el "$" como marca de ficha vacía; pero es un buen ejemplo. Utilizando este sistema, borrar la ficha 33 sería tan fácil como

```
nom$ (33)="$"
```

El problema de este sistema es que si se borran muchas fichas se desaprovecha mucho la memoria del ordenador, y si se borran muy pocas, resultará muy lento encontrar una ficha vacía.

El sistema utilizado para dar de baja debe elegirse en función del tipo de datos que vayamos a introducir en el fichero, si se van a dar muchas o pocas bajas, y de otros factores que sólo la experiencia os ayudará a discernir.



Hasta ahora ya hemos definido las operaciones fundamentales del fichero: dar de alta (añadir fichas), dar de baja (borrar fichas) y consultar. La última opción que nos queda por comentar es la búsqueda por un campo. Realizaremos la búsqueda por nombre, pero quede claro que el proceso es el mismo con cualquiera de los otros campos.

Esta opción es, con mucho, la más utilizada. Si queremos saber el teléfono de alguien, es lógico que sepamos su nombre; pero es más difícil que sepamos el número de ficha que ocupa en el fichero. Para buscar un nombre, por ejemplo "Absurdio Príntez Gotodies" tenemos que recorrer todo el fichero hasta dar con él. Podríamos programarlo de la siguiente forma.

```
    a$="Absurdio Príntez Gotodies"
    i=0
xxx i=i+1
    if (nom$ (i)<>a$ and i<=100) then
    goto xxx
yyy if (nom$ (i)=a$) then... encontrado
    else... no encontrado
```

El programa puede llegar a la línea yyy por dos razones: que haya encontrado el nombre (nom$(i)=a$) o que hayamos recorrido todas las fichas (i>100), por lo que el IF... THEN... de la línea yyy nos indica si ha encontrado o no el nombre dentro del fichero.

Llega el momento de hacer un programa que controle la totalidad del fichero. El programa del recuadro adjunto no es más que un ejemplo de uso de todo lo que hemos visto hasta el momento. Evidentemente, cada uno debe modificarlo para su aplicación particular, añadiendo o eliminando cuantas opciones desee.

DESCRIPCION DEL PROGRAMA

El programa que acompaña a este artículo no pretende, ni mucho menos, ser un buen programa de ficheros en memoria. Es tan sólo un ejemplo práctico de cómo se utilizan éstos.

Vamos, por tanto, a comentar cada uno de los bloques que lo forman.

En primer lugar, y tras las líneas REM iniciales, inicializamos los tres conjuntos que formarán nuestro fichero: nom$, edad y tlf$. También colocamos un uno en la variable NF. En esta variable almacenaremos el Número de Ficha sobre la que estamos trabajando. Al comenzar el programa colocamos un uno, ya que la primera ficha que entraremos es la ficha uno.

A partir de la línea 50 nos encontramos el menú de opciones, que nos mostrará en la pantalla todas las opciones posibles, y nos pedirá cuál queremos utilizar. Vamos a tratar ahora cada una de las opciones.

En la línea 1000 vamos a incluir la opción de DAR DE ALTA una ficha. Repasemos paso a paso el proceso a seguir. En primer lugar, cuando utilicemos esta opción, la variable NF debe contener el número de ficha a introducir. ¿Recordáis que la inicializamos con el valor uno? Pues eso quiere decir, ni más ni menos que la primera vez que utilicemos esta opción entraremos los datos de la ficha uno. En primer lugar en la línea 1030 nos encontramos un INPUT que nos pide el nombre de la ficha NF (nom$(NF)). En las dos líneas siguientes a ésta se hace lo mismo con la EDAD y el TELEFONO de esa misma ficha. A continuación, en la línea 1060, sumamos uno a la variable NF, con lo que nos preparamos para recibir la siguiente ficha. Por último una cuestión de puro trámite: preguntarnos si queremos o no entrar una nueva ficha y actuar en consecuencia.



A partir de la línea 2000 hemos incluido la opción de DAR DE BAJA una ficha. Como comentábamos anteriormente, existen muchos métodos para eliminar una ficha de un fichero en memoria; pero hemos elegido la más simple en este ejemplo. Hemos utilizado este sistema porque simplifica mucho el programa, de forma que lo puedan entender aquellos que no dominan demasiado el lenguaje BASIC. Para aquellos que dominan y conocen bien el lenguaje dejamos la labor de instalar cualquiera de los otros sistemas de borrado. El sistema utilizado consiste en preguntar el número de ficha a borrar y, tras haber hecho esto, desplazar todas las siguientes un puesto hacia atrás.

La última opción que hemos incluido en este sencillo programa es la de consultar las fichas previamente entradas. Evidentemente, antes de consultar hemos de haber entrado algunas fichas. (podéis hacer que el programa verifique este punto y lo avise si es necesario). Hemos dividido la opción en otras dos, consulta por nombre y consulta por código. En el segundo caso, en la línea 5000, no hay más que preguntar el código e imprimir todos los datos relacionados con esa ficha. Hemos incluido una verificación, para verificar que el código entrado está entre 1 y 100.

Y aquí acaba el programa. Como veis, se trata de un simple ejemplo de utilización de los ficheros en memoria; pero esperemos que sea lo suficientemente clarificador como para que podáis realizar vuestros programas de ficheros.

Por último, y antes de despedirnos vamos a haceros unas cuantas sugerencias para vuestros programas de ficheros. En primer lugar, en un programa de ficheros en memoria es muy importante verificar todos los posibles errores de los usuarios, ya que si el programa se para en algún momento pueden perderse todos los datos. Os recomendamos, por tanto, que no escatiméis sentencias del tipo IF... THEN... allí donde puedan ser interesantes. Otra recomendación interesante es que utilicéis rutinas de entradas de datos en lugar de INPUTs, ya que estos últimos producen muchos errores si el usuario del programa se equivoca. El uso y programación de las rutinas de entrada de datos fueron comentadas en el número de diciembre (25-26) de nuestra revista. Por último, utilizad todos los REMs que creáis convenientes, ya que seis meses después de haber hecho el programa los agradeceréis.

Y ahora sí, nos despedimos; pero volveremos en el siguiente número a tratar este interesante tema de los ficheros. En el próximo número: LOS FICHEROS SECUENCIALES.

```
10 '
11 ' ************************
12 ' *                      *
13 ' * Ejemplo de uso de los *
14 ' *                      *
15 ' *  ficheros en memoria  *
16 ' *                      *
17 ' * (c) MSX-EXTRA        *
18 ' *                      *
19 ' * Autor: Willy Miragall *
20 ' *                      *
21 ' ************************
22 '
23 '
25 ' ** INICIALIZO CONJUNTOS **
30 '
40 DIM NOM$(100),EDAD(100),TLF$(100)
42 NF=1
45 '
50 ' ** MENU DE OPCIONES **
55 '
60 CLS
70 PRINT "1.- DAR DE ALTA"
80 PRINT "2.- DAR DE BAJA"
90 PRINT "3.- CONSULTAR"
100 PRINT:PRINT
110 INPUT "¿Qué opción";OP
120 IF OP<1 OR OP>3 THEN GOTO 60
130 ON OP GOTO 1000,2000,3000
140 END
1000 '
1005 ' ** DAR DE ALTA **
1010 '
1020 CLS
1030 INPUT "NOMBRE";NOM$(NF)
1040 INPUT "EDAD";EDAD(NF)
1050 INPUT "TELEFONO";TLF$(NF)
1060 NF=NF+1
1070 PRINT:PRINT
1080 PRINT "¿Quieres añadir otra (S/N)
?";
1090 A$=INPUT$(1)
1100 IF A$="S" OR A$="s" THEN 1020
1110 IF A$="N" OR A$="n" THEN 50
1120 GOTO 1090
2000 '
2005 ' ** DAR DE BAJA **
2010 '
2020 CLS
2030 INPUT "Número de ficha";F
2040 FOR X=F+1 TO 100
2050   NOM$(X-1)=NOM$(X)
2060   EDAD(X-1)=EDAD(X)
2070 NEXT X
2080 PRINT "¿Quieres borrar otra (S/N)
?";
2090 A$=INPUT$(1)
2100 IF A$="S" OR A$="s" THEN 2020
2110 IF A$="N" OR A$="n" THEN 50
2120 GOTO 2090
3000 '
3005 ' ** CONSULTAR **
3010 '
3020 CLS
3030 PRINT "CONSULTA":PRINT
3040 PRINT "¿Por nombre (N) o por códi
go (C)?";
3050 A$=INPUT$(1)
3060 IF A$="N" OR A$="n" THEN 4000
3070 IF A$="C" OR A$="c" THEN 5000
3080 GOTO 3050
3100 PRINT "¿Quieres consultar otra (S
/N)?";
3110 A$=INPUT$(1)
3120 IF A$="S" OR A$="s" THEN 3020
3130 IF A$="N" OR A$="n" THEN 50
3140 GOTO 3110
4000 '
4005 ' ** CONSULTA POR NOMBRE **
4010 '
4020 PRINT
4030 INPUT "¿Qué nombre";N$
4035 PRINT
4040 I=0
4050 I=I+1
4060 IF (NOM$(I)<>N$ AND I<100) THEN G
OTO 4050
4070 IF NOM$(I)=N$ THEN PRINT "EDAD:";
EDAD(I):PRINT "TELEFONO:";TLF$(I) ELSE
 PRINT "NO ENCONTRADO"
4080 GOTO 3100
5000 '
5005 ' ** CONSULTA POR CODIGO **
5010 '
5020 PRINT
5030 INPUT "¿Qué código";C
5035 PRINT
5040 IF C>0 AND C<100 THEN PRINT "NOMB
RE:";NOM$(C):PRINT "EDAD:";EDAD(C):PRI
NT "TELEFONO:";TLF$(C) ELSE PRINT "COD
IGO NO VALIDO"
5050 GOTO 3100
```

TRUCOS DEL PROGRAMADOR



PROTECCION DE LISTADOS

Para proteger los listados de vuestros programas de miradas indiscretas podéis hacer:

```
POKE &HFF89, &HC3
POKE &HFF8A, 0
POKE &HFF8B, 0
```

Este sistema de protección consiste en eliminar la instrucción LIST, de modo que si alguien intenta hacer LIST, el ordenador hace un RESET (se apaga y vuelve a encender automáticamente).

El principal inconveniente de este sistema de protección es que, si se conoce el truco, se conoce la forma de evitarlo, ya que haciendo

```
POKE &HFF89, &HC9
```

se vuelve a conectar la instrucción LIST, y se puede ver el listado.

Sin embargo, si además de la instrucción LIST eliminamos la instrucción POKE, resulta imposible volver a activar ninguna de las dos, de modo que vuestros listados estarán definitivamente seguros.

Teclead el siguiente listado:

```
10 POKE &HFF89, &C3
20 POKE &HFF8A, 0
30 POKE &HFF8B, 0
40 POKE &HFF93, &C3
50 POKE &HFF94, 0
60 POKE &HFF95, 0
```

Tras hacer RUN podéis estar seguros de que vuestros listados serán inviolables.

TAMAÑO DE LA PANTALLA

De todos es conocida la instrucción WIDTH (o al menos debería ser conocida por todos). Esta instrucción nos permite definir el número de caracteres por línea en la pantalla. Por ejemplo, si hacemos WIDTH 15, sólo podremos escribir 15 caracteres por línea.

Sin embargo, el BASIC no nos proporciona ninguna instrucción que nos permita hacer algo simiiar con el formato vertical. Para conseguir variar el número de líneas de la pantalla basta con hacer.

POKE &HF3B1, n

donde n es el número de líneas que deseamos tenga la pantalla. Si utilizamos valores mayores que 24 puede pasar cualquier cosa. ¿Qué mejor invitación para realizar experimentos y conseguir nuevos trucos? Ya sabéis, a partir de ahora, pantallas a medida.

DEEK Y DOKE

En algunas aplicaciones (sobre todo en la programación de ensamblador desde el BASIC), resulta muy interesante hacer POKE en dos direcciones de memoria consecutivas.

Hemos definido una función que realiza esta tarea. Para utilizarla debéis hacer lo siguiente.

1.– Incluir la siguiente línea al inicio de vuestros programas.

```
10 DEF FNDOKE(X) = PEEK(X)
+256*PEEK(x+1)
```

2.– Para utilizarla no tenéis más que hacer:

```
PRINT FNDOKE (&HFF89)
ó
P=FNDOKE (64532)
etc...
```

SOUND Y LAS NOTAS

Cuando queremos utilizar la instrucción SOUND para generar notas musicales necesitaremos saber las frecuencias de cada nota. He aquí una tabla que os puede ayudar con esta tarea:

DO	-130'79
DO#	-138'50
RE	-146'78
R#E	-155'44
MI	-164'80
FA	-174'61
FA#	-184'91
SOL	-195'93
SOL#	-207'48
LA	-220
LA#	-232'98
SI	-246'94

Como dato adicional, sabed que, para pasar a una escala superior basta con multiplicar por 2 estas frecuencias, mientras que para pasar a una escala inferior basta con dividirlas por 2.





KONAMI TE PRESENTA
TRES AVENTURAS DE PELICULA
SERMA
¿HAS VISTO LAS NOVEDADES DE KONAMI?
NEMESIS, TU MEJOR JUEGO DEL ESPACIO EN MSX. SE TU EL PROTAGONISTA DE LA GUERRA DE LAS GALAXIAS. Y HABLANDO DE PELICULAS, ¿TE ACUERDAS DE LOS GOONIES?, AHORA TE TOCA A TI RESCATAR A TUS COMPAÑEROS ENCERRADOS EN LAS CUEVAS, MIENTRAS TE DESHACES DE SUS PELIGROSOS ENEMIGOS. Y POR SI TE GUSTAN LAS AVENTURAS DE TODO TIPO: NIGHTMARE, PARA QUE BUSQUES A TU PRINCESA POR LAS MIL Y UNA PANTALLAS.
NEMESIS
Knightmare
©Konami 1986
魔城伝説
The GOONies
©Konami 1986
グーニーズ
P.V.P : 5.200 pts.
RECORTA Y ENVIA ESTE CUPON A: SERMA. C/. CARDENAL BELLUGA, 21. 28028 MADRID. TELFS. 256 21 01./02.
TITULO:
NOMBRE Y APELLIDOS:
DIRECCION:
CODIGO POSTAL:
POBLACION:
PROVINCIA:
FORMA DE PAGO:
ENVIO TALON BANCARIO ☐ - CONTRA REEMBOLSO ☐